Human Activity Recognition with Wearable Sensors

A dissertation submitted to
TECHNISCHE UNIVERSITÄT DARMSTADT
Fachbereich Informatik

for the degree of
Doktor-Ingenieur (Dr.-Ing.)

presented by

DUY TAM GILLES HUYNH
Dipl. Inform.

born 29th of July, 1975
in Paris, France

Prof. Dr. Bernt Schiele, examiner
Prof. Dr. Paul Lukowicz, co-examiner

Darmstadt, 2008
D17

Date of Submission: 16th of June, 2008
Date of Defense: 25th of August, 2008

Abstract

This thesis investigates the use of wearable sensors to recognize human activity. The activity of the user is one example of *context information* – others include the user's location or the state of his environment – which can help computer applications to adapt to the user depending on the situation. In this thesis we use wearable sensors – mainly accelerometers – to record, model and recognize human activities. Using wearable sensors allows continuous recording of activities across different locations and independent from external infrastructure. There are many possible applications for activity recognition with wearable sensors, for instance in the areas of healthcare, elderly care, personal fitness, entertainment, or performing arts.

In this thesis we focus on two particular research challenges in activity recognition, namely the need for less supervision, and the recognition of high-level activities. We make several contributions towards addressing these challenges. Our first contribution is an analysis of features for activity recognition. Using a data set of activities such as *walking*, *standing*, *sitting*, or *hopping*, we analyze the performance of commonly used features and window lengths over which the features are computed. Our results indicate that different features perform well for different activities, and that in order to achieve best recognition performance, features and window lengths should be chosen specific for each activity.

In order to reduce the need for labeled training data, we propose an unsupervised algorithm which can discover structure in unlabeled recordings of activities. The approach identifies correlated subsets in feature space, and represents these subsets with low-dimensional models. We show that the discovered subsets often correspond to distinct activities, and that the resulting models can be used for recognition of activities in unknown data. In a separate study, we show that the approach can be effectively deployed in a semi-supervised learning framework. More specifically, we combine the approach with a discriminant classifier, and show that this scheme allows high recognition rates even when using only a small amount of labeled training data.

Recognition of higher-level activities such as *shopping*, *doing housework*, or *commuting* is challenging, as these activities are composed of changing sub-activities and vary strongly across individuals. We present one study in which we recorded 10h of three different high-level activities, investigating to which extent methods for low-level activities can be scaled to the recognition of high-level activities. Our results indicate that for settings as ours, traditional supervised approaches in combination with data from wearable accelerometers can achieve recognition rates of more than 90%.

While unsupervised techniques are desirable for short-term activities, they become crucial for long-term activities, for which annotation is often impractical or impossible. To this end we propose an unsupervised approach based on topic models that allows to discover high-level structure in human activity data. The discovered activity patterns correlate with daily routines such as *commuting*, *office work*, or *lunch routine*, and they can be used to recognize such routines in unknown data.

Zusammenfassung

Diese Arbeit untersucht den Einsatz von tragbaren Sensoren zur Erkennung menschlicher Aktivitäten. Die Aktivitiät des Benutzers ist ein Beispiel von *Kontext-Information* – andere Beispiele sind der Aufenthaltsort des Benutzers, oder der Zustand seiner Umgebung. Die Erkennung von Kontext ermöglicht es Applikationen, sich an die Situation des Benutzers anzupassen. In dieser Arbeit verwenden wir tragbare Sensoren – hauptsächlich Beschleunigungssensoren – um menschliche Aktivitäten aufzunehmen, zu modellieren und zu erkennen. Der Einsatz von tragbaren Sensoren ermöglicht eine kontinuierliche Aufnahme, unabhängig von externer Infrastruktur. Für die automatische Erkennung von Aktivitäten existiert eine Vielzahl von Einsatzfeldern, beispielsweise im Gesundheitswesen, bei der Altersfürsorge, im Fitness-Bereich, oder im Unterhaltungsbereich.

In dieser Arbeit konzentrieren wir uns auf zwei spezielle Herausforderungen bei der Erkennung von Aktivitäten: Erstens die Notwendigkeit, den Umfang der Überwachung beim Training von Algorithmen des maschinellen Lernens zu reduzieren. Zweitens die Erkennung von höherwertigen Aktivitäten, die sich über längere Zeiträume erstrecken und aus mehreren Aktivitäten zusammengesetzt sein können. Zum Erreichen dieser Ziele macht diese Arbeit mehrere Beiträge. Den Anfang macht eine Analyse von Merkmalen (Features) für die Erkennung von Aktivitäten. Anhand eines Datensatzes von Aktivitäten wie *Laufen*, *Sitzen*, *Stehen*, oder *Springen* analysieren wir die Effizienz verschiedener gebräuchlicher Merkmale und Fensterlängen, über welche die Merkmale berechnet werden. Unsere Resultate deuten darauf hin, dass die Performanz einzelner Merkmale von der zu erkennenden Aktivität abhängt, und dass die besten Resultate dann erzielt werden, wenn Merkmale und Fensterlängen pro Aktivität individuell ausgewählt werden.

Um die Notwendigkeit von annotierten Trainingsdaten zu reduzieren, schlagen wir einen unüberwachten Lernalgorithmus vor, der Struktur in nicht-annotierten Aufnahmen von Aktivitiäten entdecken kann. Der Ansatz identifiziert korrelierte Untermengen des Merkmals-Raums und repräsentiert diese mit niedrig-dimensionalen Modellen. Wir zeigen, dass die entdeckten Untermengen oft mit unterschiedlichen Aktivitäten übereinstimmen, und dass die resultierenden Modelle zur Erkennung von Aktivitäten in unbekannten Daten eingesetzt werden können. In einer weiteren Studie zeigen wir, dass der Ansatz für semi-überwachtes Lernen (semi-supervised learning) verwendet werden kann. Genauer gesagt kombinieren wir den Ansatz mit einem diskriminanten Klassifizierer, und zeigen, dass dieser Ansatz zu hohen Erkennungsraten führt, selbst wenn nur kleine Mengen an annotierten Trainingsdaten verwendet werden.

Die Erkennung von höherwertigen Aktivitäten wie *Einkaufen*, *Hausarbeiten verrichten*, oder *(zur Arbeit) Pendeln* ist eine Herausforderung, da diese Aktivitäten sich aus wechselnden Unteraktivitäten zusammensetzen, und von Person zu Person stark variieren. Wir führen eine Studie mit drei verschiedenen Typen von höherwertigen Aktivitäten durch, in der wir untersuchen, zu welchem Ausmaß sich traditionelle Methoden zur Aktivitätenerkennung auf solche Aktivitäten anwenden lassen. Die Resultate deuten an, dass sich unter bestimmten Bedingungen Erkennungsraten von mehr als 90% erreichen lassen.

Während unüberwachte Methoden für kurzfristige Aktivitäten vorteilhaft sind, sind sie für längerfristige und höherwertige Aktivitäten unabdingbar, da für solche Aktivitäten Annotationen nur sehr schwer zu erlangen sind. Zu diesem Zweck schlagen wir eine unüberwachte Lernmethode vor, die Struktur in höherwertigen Aktivitätsdaten aufdeckt. Die extrahierten Aktivitätsmuster korrelieren mit alltäglichen Routinen wie *Pendeln*, *Büroarbeit*, oder *Mensa-Routine*, und sie können zur Erkennung solcher Routinen in unbekannten Daten eingesetzt werden.

Acknowledgments

First I would like to sincerely thank my supervisor, Professor Bernt Schiele, for giving me the opportunity to enter the world of research, and for providing guidance, motivation and countless advice during my academic journey. I'm amazed that I got this far, and it would not have been possible without his excellent support. Thanks, Bernt! Apart from that, I would also like to thank Professor Paul Lukowicz for agreeing to be the co-examiner of this thesis.

Having a great supervisor is important, but equally important is to have great colleagues, and there is certainly no lack of them at the MIS group in Darmstadt. First I would like to thank my longtime office mate Nicky Kern, for introducing me to the details of life as a researcher in wearable computing, as well as for his countless advice, support and his amazing and contagious cheerfulness. Thanks also to my other office mates Victoria Carlsson, Andreas Zinnen, Ulf Blanke, Ulrich Steinhoff and Maja Stikic, for stimulating discussions and valuable feedback on my research and on other aspects of life. Furthermore, I'd like to thank former group members Stavros Antifakos and Florian Michahelles, who introduced me to their projects at ETH Zurich and gave me a key motivation to apply to the group in the first place.

Another important source of inspiration (and fun) were our bi-annual retreats, which provided plenty of input and stimulating discussions from all group members. Apart from the ones I already mentioned above, these are Kristof van Laerhoven, Edgar Seemann, Bastian Leibe, Michael Stark, Micha Andriluka, Gyuri Dorkó, Krystian Mikolajczyk, Nikodem Majer, Christian Wojek, Paul Schnitzspahn, and Stefan Walk.

Several colleagues have contributed to specific parts of this thesis. Major thanks go to Kristof Van Laerhoven and his team of students, who provided me with several versions of his wearable sensor platform, allowing me to concentrate on the algorithmic challenges of the thesis. Mario Fritz provided valuable support and insight for our work on applying topic models to data from wearable sensors, which is described in Chapter 7. Ulf Blanke was of crucial help during our first study on high-level activities described in Chapter 6. The recordings of the study were created during his diploma thesis, which I supervised. Thanks also to Bastian Leibe for insightful advice on applying the multiple eigenspace approach to activity data (Chapter 4).

Part of this work has been funded by the MOBVIS project (*Vision Technologies and Intelligent Maps for Mobile Attentive Interfaces in Urban Scenarios*), funded by the European Commission (FP6-511051). At this point I'd also like to thank the members of the MOBVIS team for many interesting meetings and discussions in various cities across Europe, and in particular Aleš Leonardis for advice on the use of the multiple eigenspace approach.

Finally I'd like to thank my family for their support over the course of the thesis and beyond. I consider myself very lucky to have enjoyed such a happy family life in parallel to my work at university.

Contents

1

Introduction

Computers are becoming more *pervasive* - they are embedded in our phones, music players, cameras, in clothing, in buildings, cars, and in all kinds of everyday objects which do not resemble our long-established image of a desktop PC with a screen, keyboard and mouse. How should we interact and live with many computers that are small, and sometimes hidden so that we cannot even see them? In which ways can they make our lives better? The vision of *ubiquitous computing* is that, eventually, computers will disappear and become part of our environment, fading into the background of our everyday lives [Weiser 1991]. Ideally, there will be more computers, invisibly enhancing our surroundings, but we will be less aware of them, concentrating on our tasks instead of on the technology.

As designers of ubiquitous computing technologies, we are challenged to find new ways to interact with this new generation of computers, and new uses for them. One way of making computers disappear is to reduce the amount of *explicit interaction* that is needed to communicate with them, and instead increase the amount of *implicit interaction*. A very simple example of implicit interaction would be the following: instead of pushing a switch to turn on the light, we could use a motion sensor that registers when someone enters a room and turns the light on. Thus, we have reduced the amount of explicit action a user must take, and instead used a *sensor* to control the light source with a simple rule.

Thinking further about the light switch example, we would soon notice that simply turning the light on when motion is detected in a room would not be optimal for several reasons. For example, the lights would probably turn off if someone was reading and not moving for some time, and they would turn on even when enough sunlight entered through the windows. Sometimes the inhabitant might not even want the lights to be turned on, e.g. when resting or sleeping. Thus, in order to best serve its purpose, an ideal system in this case would need to have a very good understanding of the state of the user and his environment, i.e. of the user's *context*.

Typical information used to describe a user's context include his identity, his current location, the activity he is currently performing, his social interactions or the state of his environment. Research in *context-aware computing* uses sensors in the environment or

carried or worn by the user to extract and interpret the user's context. By using contextual information, the physical world surrounding the user becomes part of an application's interface, which might reduce or even eliminate the need for explicit user input, thus allowing the user to concentrate on his task rather than on the computer interface.

This thesis explores a sub-field of context-aware computing. First, we concentrate on a single yet crucial part of the user's context, namely his or her current *activity*. Second, we use *wearable sensors* combined with methods from *machine learning* to record, model, learn and recognize the user's activity. One of the key benefits of wearable systems is the opportunity to perceive the world from a first-person perspective, continuously, and without the need of external infrastructure. As we will see in the next chapter, activity recognition with wearable sensors has the potential to enhance existing applications as well as enable new ones, ranging from personal healthcare and assisted living, to industrial applications, and even entertainment and arts.

Next, we present the main challenges that researchers face when dealing with activity recognition, and then give an overview of the contributions that this thesis makes in addressing these challenges.

1.1 Challenges

In the following we discuss important research challenges in activity recognition. The first two of them are central to this thesis – namely the constant need for less supervision (Section 1.1.1), and the exploration of long-term and high-level activities (Section 1.1.2). We start by discussing these two challenges in more detail, and then give an overview of other challenges of the field.

1.1.1 The need for less supervision

Most approaches in activity recognition rely on annotated recordings of activity data in order to train a machine learning algorithm. Obtaining such data, especially together with sufficiently detailed annotations, or *ground truth*, is tedious, time-consuming, erroneous, and may even be impossible in some cases, thus posing a significant barrier to progress in the field. Due to the difficulties involved in obtaining annotated training data, a large part of the work that has been published so far is based on data from few individuals, often the researchers themselves. This leads to various problems, the first and foremost being that the amount of data is often too small to make reliable estimations of how well the approach will generalize to different settings or users, or in other words, how useful the approach is.

Being able to exploit and learn from non- or sparsely-annotated data, which can be obtained much easier, would greatly simplify many problems in the field of activity recognition. Moreover, there exist large databases of unlabeled data, such as cell phone logs,

location traces from navigation systems, or even activity diaries from social networking communities and blogs, which contain a wealth of information from an activity recognition point of view. For such large databases, any annotation is expensive, or may even introduce unforeseen bias in the data. As a consequence, semi-supervised and unsupervised machine learning methods that can learn from unlabeled data become more and more important. Other fields have already realized the need to adopt methods that can acommodate for large unlabeled data collections, and the activity recognition community could greatly benefit from such methods as well.

1.1.2 Long-Term and High-Level Activities

As we will see in the next chapter, a majority of work in activity recognition focuses on relatively short activities that can be measured in minutes or seconds, rather than hours or even days. Exploring activities on larger time scales is interesting and challenging for several reasons. First, while activity recognition on small time scales is still poorly understood in many respects, long-term and high-level activities have been even less explored. E.g., there is neither a consensus about how to define high-level activities, nor which type of algorithms to use to model, learn and recognize them.

On larger time scales, the types and properties of activities are different from those on smaller scales, which poses challenges to existing recognition approaches. For instance, activities such as *working* or *going shopping* consist of many sub-activities, which are performed in changing order and can possibly overlap. They also exhibit a larger variance in execution than short activities (e.g., think of *shopping in a supermarket* and *strolling through a pedestrian zone* as two different instances of *going shopping*, or of *office work* and *construction work* of two different *working* activities). Location and time of day become more important, but are often not enough to reliably characterize an activity: there is a large variability in human activities, and different activities can be performed at the same location (e.g. *holding a meeting in a restaurant*, *having lunch at the office desk*, etc.).

In addition to these challenges, long-term recordings require new annotation techniques that minimize the burden on the user while still attaining sufficiently detailed ground truth. Moreover, they cannot be deployed in laboratory settings but must be conducted "in the wild", i.e. in everyday environments, which requires robust and power-efficient hardware. Finally, they require efficient algorithms in order to deal with possibly large amounts of data.

1.1.3 Other Challenges

Recognition of Short Activities. Recognizing very short physical activities, for instance hand-gestures for controlling an application, is challenging for several reasons.

E.g., when aiming to 'spot' and recognize such gestures in continuous and long recordings, they must first be separated from a possibly large amount of irrelevant background data. Another challenge is the fact that such activities might be performed in parallel with other activities, e.g. while jogging or riding a bike [Zinnen and Schiele 2008, Junker *et al.* 2008].

Lack of Reference Data and Evaluation Procedures. A common problem in the activity recognition community is the lack of annotated reference data that could help researchers to compare the performance of their approaches. Efforts in this direction have been made [Junker *et al.* 2004], but they haven't caught on widely yet. The lack of standard evaluation procedures is related to this problem, which has been addressed eg. by [Ward *et al.* 2006b, Minnen *et al.* 2006a]. Evaluation is particularly difficult for unsupervised approaches for which no ground truth is available. Organizing competitions on reference data sets, which is common practice in other fields such as computer vision, might be one way of motivating researchers to agree on standards.

Hardware Challenges. In the area of sensor platforms, miniaturization of components, as well as increases in memory size and processing power are quickly progressing. However, energy supply is a bottleneck, which is a problem in particular for continuous and long-term recognition of activities. This problem has been addressed e.g. by using large amounts of very simple and low-power sensors [Wren *et al.* 2007] or by cleverly switching between high-frequency, yet power hungry, and low-frequency, yet power efficient, types of sensors, depending on the activity level of the user [Van Laerhoven *et al.* 2006]. Reliability and robustness are also challenges for hardware design, especially in the wearable context.

New Types of Sensors. Human activities are so diverse that there does not exist one single type of sensor that could recognize all of them. For example, while physical motion can be well-recognized by inertial sensors, other activities, such as talking, reading, chewing, or physiological states of the user, can be better recognized with other, sometimes dedicated sensors. Making sensors less obtrusive, more robust, easier to use, washable, even attractive, are other challenges which are addressed e.g. by [Buechley and Eisenberg 2007].

Privacy and Security. In order for activity recognition and ubicomp technologies in general to become widely used and accepted, the prospective users must be sure that their privacy concerns are respected. For instance, on an application level, it is important to make sure that no data is recorded without the user's consent, that devices feature an 'off' or 'mute' switch, and that the recorded data and any data derived from it stays under the control of the user. However, privacy aspects can already be considered in the design of ubiquitous computing systems, leading to various research challenges [Langheinrich

2005]. Security is a related topic, which has been addressed e.g. by [Mayrhofer and Gellersen 2007]. Another related topic

Platforms and Toolkits for Soft- and Hardware. Being able to make use of existing hard- and software platforms is important for researchers to avoid reinventing the wheel and instead focus on the interesting problems. Especially hardware development can be time-consuming, and being able to use available sensor platforms such as [Choudhury *et al.* 2008] or [Van Laerhoven *et al.* 2006] can save valuable time. Much effort has gone into development of hardware toolkits (e.g. [Greenberg and Fitchett 2001, Holmquist *et al.* 2003, Buechley 2006]) and software toolkits (e.g. [Salber *et al.* 1999, Hartmann *et al.* 2007]). These can also help in teaching classes about wearable and context-aware computing, thus introducing these fields to a wider audience of possible future researchers.

1.2 Contributions

In the following we summarize the contributions that this thesis makes. Overall, the topics investigated in this thesis center around two main themes, which can be formulated in the form of two questions:

1. How can we reduce the amount of annotation needed for activity recognition?
2. How can we recognize high-level behavior from low-level sensors?

The thesis takes several steps towards answering these questions, starting with several separate investigations, and concluding by proposing a joint approach, namely an unsupervised method for discovering high-level structure in daily activities. In the following we describe our contributions in more detail.

We begin our investigations with ***a study of features for activity recognition***. The choice of features is a fundamental first step in applying machine learning methods to sensor data, and it can have a strong influence on the outcome of any approach. Most existing approaches use a fixed set of features, regardless of the activity to be recognized. In Chapter 3, we show that recognition rates can be improved by careful selection of individual features for each activity. The main contribution of the chapter is a systematic analysis of features computed from a real-world data set, showing how the choice of feature and the window length over which the features are computed affects the recognition rates for different activities.

In Section 1.1.1, we have argued that unsupervised techniques for activity recognition are highly desirable, since they allow us to learn from non- or sparsely-annotated data. To this end, Chapter 4 proposes ***a novel approach to discover structure in sensor data***

of human physical activity. We demonstrate the feasibility of the approach by applying it to acceleration data recorded from body-worn sensors. When applied to this data, our approach is able to build low-dimensional models that correspond to different activities, without requiring any prior training, user annotation or information about the number of activities involved. When used for classification, the system shows recognition rates comparable to other, supervised approaches. Also, we show that classification rates can be improved when combining the data of two sensors located at different body locations.

In Chapter 5 ***we investigate the topic of semi-supervised learning for activity recognition***. We use the unsupervised and generative approach of Chapter 4 and show that combining it with a discriminative classifier yields several advantages. More specifically, the generative part of the algorithm allows to extract and learn structure in activity data without any prior labeling or supervision. The discriminant part then uses a small but labeled subset of the training data to train a discriminant classifier. In experiments we show that this scheme enables to attain high recognition rates even though only a subset of the labeled data is used for training. In addition to this, we present an analysis and discussion of the tradeoff between labeling effort and recognition performance.

In Chapter 6 we make a first step towards recognition of high-level activities as they occur in daily life. More specifically, ***we investigate how far existing methods for the recognition of low-level activities can be scaled to the recognition of high-level activities***. To this end, we use a 10h data set to analyze the performance of four different algorithms for the recognition of both low- and high-level activities. Our experimental results suggest that it is feasible to recognize certain high-level activities, such as *shopping* or *doing housework*, using the same algorithms as for the recognition of low-level activities.

At the end of the thesis we bring together its two main themes, namely the quest for less supervision and the investigation of high-level activities. In Chapter 7, we introduce ***a novel approach for modeling and discovering daily routines*** such as *commuting*, *office work*, or *lunch routine*, from on-body sensor data. We propose to model such routines as a probabilistic combination of activity patterns. The use of topic models enables the automatic discovery of such patterns in a user's daily routine. In addition, we report experimental results that show the ability of the approach to discover and model daily routines without user annotation.

Parts of this thesis have been published in refereed conference and workshop papers. The feature analysis presented in Chapter 3 is reported in [Huỳnh and Schiele 2005]. The unsupervised approach for modeling activities (Chapter 4) was first proposed in [Huỳnh and Schiele 2006b]. The approach is extended in combination with a discriminative classifier in [Huỳnh and Schiele 2006a] (Chapter 5). Recognition of high-level activities is addressed in [Huỳnh *et al.* 2007] (Chapter 6). The approach for discovering daily routines (Chapter 7) is introduced in [Huỳnh *et al.* 2008]. In addition, the author has contributed the activity recognition part of the hybrid approach presented in [Stikic *et al.* 2008a].

1.3 Thesis Outline

This thesis is organized as follows.

Chapter 2 gives an overview of related work in the area of activity recognition with wearable sensors. We give a brief historical perspective and then review different applications, sensors, activities and machine learning approaches that have been proposed.

Chapter 3 presents a systematic comparison of commonly used features for activity recognition. The choice of features can be crucial for the success of a recognition algorithm, and prior to our study there existed little work on evaluation of features with respect to recognition of specific activities.

Chapter 4 introduces a novel approach for unsupervised learning of activities from low-level sensor data. We describe the algorithm, propose an extension to multiple time scales, and evaluate the approach on several data sets, showing that it can be used to reliably model and also recognize activities.

Chapter 5 suggests a strategy for combining generative and discriminative models in order to reduce the amount of labeled training data while leveraging information from unlabeled samples. This is achieved by combining the unsupervised approach that we introduced in Chapter 4 with the discriminative power of support vector machines.

Chapter 6 takes a step towards recognition of high-level activities from body-worn accelerometers. We explore to which extent existing methods for activity recognition can be scaled to more complex and long-term activities such as *going shopping* or *doing housework*.

Chapter 7 continues on the path towards modeling and recognition of high-level activities. We introduce an unsupervised approach that allows us to discover and later recognize daily routines such as *working in the office* or *commuting* from body-worn accelerometers.

Chapter 8 summarizes the work of the thesis, draws conclusions and gives an outlook to possible future work.

2

Related Work

This chapter gives an overview of the state of the art, current topics and challenges in activity recognition using wearable sensors. One of the main promises of wearable computing is to enable personal applications that can adapt and react to the current context of the user. While the term *context* is usually broadly defined and can in principle encompass any kind of information that relates to the current situation of the user or the objects surrounding him [Dey 2001], this thesis focuses on the user's *activity*, which is often considered one of the most important ingredients of context besides the user's location. Thus in the following we give an overview of work that uses wearable sensors to infer the current activity of the user.

The chapter is organized as follows. In the next section we give a brief historical perspective on wearable computing in general and on activity recognition in particular. Section 2.2 highlights different application areas, and Section 2.3 gives an overview of the different types of sensors that have been used for actvity recognition. Section 2.4 describes different types of activities, and Section 2.5 introduces the machine learning approaches that have been applied for activity recognition.

2.1 Introduction

Early work in activity recognition with wearable sensors dates back to the 1990s, when advances in hardware technology made sensing-, display- and computing-equipment lightweight enough so that an integrated mobile system could be "worn" by a single person for an extended period of time (e.g. as described in [Starner *et al.* 1999]). Although these research prototypes were still relatively bulky and a long way from "vanishing into the background" as envisioned by Mark Weiser [Weiser 1991], they held the exciting promise of making the computer perceive human life from a first-person perspective, thus enabling truly "personal" applications. Early work centered on traditional, text- and keyboard-based applications, and then gradually explored new methods of input and interaction e.g. using wearable cameras [Schiele *et al.* 1999, Starner *et al.* 1997] or microphones [Clarkson and Pentland 1998] and incorporating other context information such as the user's

current location, the subject of a conversation or the identity of a conversation partner in order to provide the user with relevant information about his current situation in real-time, or to store information for later retrieval [Rhodes 1997].

Measuring the physical activity of a person through the use of objective technology has been a longstanding goal of the medical research community, and accelerometers have been used for this purpose since several decades (e.g. [Montoye *et al.* 1983, Wong *et al.* 1981]). These early systems aimed to estimate global measures such as the total energy expenditure or the oxygen requirement of the subject while he or she was performing a number of different activities. Mobile systems incorporating inertial sensors that could separate and recognize specific physical activities emerged at the turn of the last decade, stimulated both by advances in hardware technology, machine learning methods, and by their expected usefulness for the new paradigm of context-aware computing (e.g. [Golding and Lesh 1999, Randell and Muller 2000, Cakmakci and Van Laerhoven 2000]).

Current research in activity recognition from wearable sensors covers a wide range of topics, with research groups focusing on topics such as the recognition of activities of daily living (ADLs) in the context of healthcare and elderly care (e.g. [Lester *et al.* 2006]), automated discovery of activity primitives in unlabeled data (e.g. [Minnen *et al.* 2006b]), semi- or unsupervised learning of activities (e.g. [Wyatt *et al.* 2005, Huỳnh and Schiele 2006b]), or the combination of several sensor modalities to improve recognition performance (e.g. [Stiefmeier *et al.* 2006, Wang *et al.* 2007]).

2.2 Applications

In the following we outline application areas for activity recognition systems in wearable or mobile settings. We begin with applications for healthcare and assisted living, which represent an important class of applications. Besides that, there exist a number of other application areas, such as industrial applications and applications for entertainment and gaming, which we outline afterwards.

Healthcare and Assisted Living. A major goal of current research in activity recognition and context-aware computing in general is to enable new health-related applications and technologies for the aging. Longer life expectancy and declining fertility rates are increasing the proportion of the elderly population in societies worldwide and posing challenges to existing healthcare systems. It is hoped that technology can help in addressing these challenges, for instance by helping elderly people to live more independent lives and thus reducing the burden of care-givers.

One type of system designed for elderly people aims to detect potentially dangerous situations in a person's life in order to call for external help automatically. Such systems can be seen as a complement to traditional emergency systems such as smoke- or fire-alarms, by detecting e.g. when a person has fallen (e.g. [Jafari *et al.* 2007, Bourke *et al.*

2007]), or when vital body signs indicate imminent health threats (e.g. [Liszka *et al.* 2004, Villalba *et al.* 2006, Anliker *et al.* 2004]).

Preventing age-related diseases or severe medical conditions before they actually happen is the goal of another class of applications, which employ long-term monitoring to detect changes or unusual patterns in a person's daily life that may indicate early symptoms of diseases such as Alzheimer's. While automatic detection of subtle behavioral changes is highly challenging and still a long-term goal of current research, applications that accumulate and summarize statistics about daily activities (e.g. [Choudhury *et al.* 2006]) or perform continuous recordings of physiological parameters (e.g. [Van Laerhoven 2004, Anliker *et al.* 2004, Liszka *et al.* 2004, Paradiso *et al.* 2005]) can already be valuable for physicians and care-givers to estimate the physical well-being of a person.

A third type of health-related system aims to use context-information to promote a more active and thus healthy lifestyle, or to actively support elderly or disabled people in performing everyday activities. E.g., [Maitland *et al.* 2006] use fluctuations in mobile phone signals to estimate and summarize a person's activity levels in order to motivate and encourage reflection on daily activities. [Consolvo *et al.* 2008] pursue similar goals by using wearable sensors to recognize specific activities and representing them as different kinds of flowers in a mobile phone display. [Andrew *et al.* 2007] combine activity and location information from wearable sensors to suggest spontaneous exercises, e.g. by noting that the user has enough time to walk to the next bus stop instead of waiting at the current one. [Dunne *et al.* 2006] use wearable optical sensors to monitor spinal posture, e.g. to detect and prevent back problems due to poor posture. [Patterson *et al.* 2004] propose a system for mentally disabled people that analyzes a user's location traces, detects anomalies (e.g. when the user is likely to have taken the wrong bus) and aids in navigation (e.g. by telling the user where to get off and which bus to take next). [Backman *et al.* 2006] and [Si *et al.* 2007] aim to support persons suffering from dementia through the use of context-aware reminders and similar assistance. [Brashear *et al.* 2003] propose a mobile system for recognition of sign-language based on wearable cameras and accelerometers.

Industrial Applications. In mobile industrial settings, activity-aware applications have the potential to support workers in their tasks, help to avoid mistakes and increase workplace safety, for instance. Wearable platforms supporting workers in tasks such as communication, access to information, or data collection, have been commercially available since the early 1990s from companies such as Xybernaut [Xybernaut Corp. 2008]. According to an overview by [Stanford 2002], early adopters of such (costly) systems were companies in which mobile knowledge workers construct, maintain and repair technically complex and costly systems such as ships, airliners and telecommunication networks.

Currently several research groups explore the next generation of industrial applications which, among other improvements, take better advantage of the multimodal sensing capabilities of wearable platforms by inferring context information such as the user's current activity. For instance, [Lukowicz *et al.* 2007] investigate the use of wearable

computing technology for scenarios in aircraft maintenance, car production, hospital environments and emergency response. In these scenarios, wearable technology and activity recognition are used to provide interactive and hands-free access to information such as electronic manuals or patient records, assist in training of new workers, provide summaries of performed activities, as well as to help in navigation and communication.

[Stiefmeier *et al.* 2008] use information gathered from wearable and environmental sensors for tracking activities of workers in car manufacturing plants, e.g. to provide realtime feedback to the worker about upcoming assembly steps or to issue warnings when procedures are not properly followed. [Ward *et al.* 2006a] combine data from wearable microphones and accelerometers in order to track wood shop activities such as sawing or hammering.

[Bardram and Christensen 2007] and [Tentori and Favela 2008] report on projects aimed at supporting hospital staff in their daily routines, e.g. by displaying health records of nearby patients on a mobile display, prioritizing patient care based on the patient's health condition, maintaining awareness of the patients status, and improving communication between patients and nurses. E.g., [Tentori and Favela 2008] envision a bracelet worn by nurses, fitted with LEDs for each patient that change color based on the patients health conditition.

Pentland et al. use wearable and context-aware computing to analyze social patterns in organizations, thereby extending techniques such as activity-recogniton to possibly large networks of individuals [Pentland 2007, Pentland *et al.* 2005]. For instance, they analyze face-to-face conversations using wearable audio (described in [Choudhury and Pentland 2003]), in order to map social networks, identify experts in the organization and help to put together project teams. In [Eagle and Pentland 2006a], they use context information gathered from mobile phones in order to identify common structures in the users' daily routines.

Entertainment and Games. Wearable systems using activity recognition are appealing for applications in the performing arts, e.g. by allowing dancers to augment their performance with interactive multimedia content that matches their motions. Such systems are described e.g. by [Aylward *et al.* 2006, Enke 2006, Barry *et al.* 2005], who employ wearable inertial sensors combined with machine learning techniques in order to record, classify and visualize the motion of dancers. For entertainment and gaming systems in general, the adoption by users may be faster than in other domains, since classification accuracy is less crucial than e.g. for healthcare systems, and since they usually raise less privacy concerns. Two out of many examples of gaming applications are the system described by [Zhang and Hartmann 2007], in which a motion-sensing clamp attached to the body or other objects is used to control video-games, or the system by [Heinz *et al.* 2006], in which wearable inertial sensors are used to recognize moves to control martial arts games. The recent popularity of game controls based on accelerometers, sparked by systems such as Nintendo's Wii platform [Nintendo 2008] or the Apple iPhone [Apple Inc. 2008], has introduced a wide audience to ideas that originated in the context-aware

computing research community and are now being widely adopted by companies and independent developers.

Other Application Areas. There exist numerous other possible application areas for wearable computing combined with activity recognition, of which the following examples should provide a brief impression. For example, [Sala *et al.* 2007] explore the use of activity-recognition for mobile context-aware advertising. In an educational context, [Beaudin *et al.* 2007] investigate the use of wearable RFID readers combined with tagged objects for casual learning of a foreign language vocabulary. Finally, [Minnen *et al.* 2007] use a wearable sensing platform to categorize soldier activities, in order to automatically compile action reports or help in recalling situations during missions.

2.3 Sensors

The types of sensors used for activity recognition range from relatively simple sensors with discrete output, such as ball switches [Van Laerhoven and Gellersen 2004] or RFID tag readers (e.g. [Patterson *et al.* 2005, Philipose *et al.* 2004]), to sensors with continuous output such as accelerometers (e.g. [Bao and Intille 2004, Cakmakci and Van Laerhoven 2000]), to more complex sensing methods such as audio processing (e.g. [Choudhury and Pentland 2003, Stäger *et al.* 2004]) and computer vision (e.g. [Nowozin *et al.* 2007, Jebara and Pentland 1999, Shi *et al.* 2004]).

Other, less commonly used types of sensors that have been proposed include fiber-optical sensors to measure posture [Dunne *et al.* 2006], foam pressure sensors to measure respiration rate [Brady *et al.* 2005], force sensitive resistors to measure muscle contractions [Lukowicz *et al.* 2006], and various kinds of physiological sensors such as oximetry sensors [Oliver and Flores-Mangas 2006], skin conductivity sensors [Westeyn *et al.* 2006], electrocardiographs [Linz *et al.* 2006], body temperature sensors, and combinations of these (e.g. [Gerasimov 2003]).

Accelerometers are probably the most commonly used type of sensor for activity recognition with wearable sensors. Besides the fact that they usually lead to good results in recognition of physical activities, they are small and cheap, require relatively little energy, memory and processing power, and are fairly insensitive to environmental conditions. In addition, users sometimes consider them less intrusive than other sensors such as microphones or cameras.

Another approach for activity recognition is based on object use, as demonstrated for instance by [Philipose *et al.* 2004, Patterson *et al.* 2005, Naeem *et al.* 2007]. These authors instrument objects in the environment with RFID tags, and use data from a wearable RFID tag reader to infer household activities (such as preparing food, doing laundry, washing dishes, etc.). Some of these methods based on Dynamic Bayesian Networks are

very flexible in principle, although the use of RFID tags restricts the approaches to closed instrumented environments.

Depending on the type of activity, recognition performance can be improved by using the same type of sensor at multiple body locations (e.g. multiple accelerometers as used by [Van Laerhoven and Gellersen 2004, Bao and Intille 2004, Huỳnh *et al.* 2007]), employing networks of heterogeneous sensors (e.g. [Junker *et al.* 2003, Kern *et al.* 2004]) or integrating a variety of sensors on a single device (e.g. [Choudhury *et al.* 2008]). Combining two or more complementary types of sensor data can also help in recognizing activities, e.g. by combining motion- and audio-data (e.g. [Lukowicz *et al.* 2004, Kern *et al.* 2004, Choudhury and Pentland 2003]), motion- and proximity-data [Stiefmeier *et al.* 2006], motion- and location-data (e.g. [Subramanya *et al.* 2006]), or motion-data and readings from wearable RFID tag readers (e.g. [Wang *et al.* 2007, Stikic *et al.* 2008a]). The latter is an example of combining wearable sensors with an instrumented environment (in this case RFID-tagged objects). A similar approach is taken by [Stikic *et al.* 2008b], who combine data from wearable accelerometers and environmental infra-read sensors. The data used in their approach is part of a larger data set introduced by [Logan *et al.* 2007], who use wearable sensors in combination with a range of environmental sensors for detecting object usage (e.g. reed switches and motion sensors for detecting usage of doors, windows, cabinets, etc.) and environmental conditions (light, temperature, humidity, etc.).

2.4 Activities

With so many envisioned applications and types of sensors to choose from, it comes as no surprise that the list of activities that people have tried to recognize with wearable sensors is long. We have already mentioned a number of activities during the discussion of applications in Section 2.2, thus we only give a brief overview in the following. We then discuss in more detail related work on high-level activities, which is the focus of Chapter 6 and Chapter 7 of this thesis.

Physical activities such as *walking*, *standing*, *sitting* and *jogging* naturally lend themselves to recognition with inertial sensors, as these activities are clearly defined by the motion and relative positions of the user's body parts. 2D- and 3D-acceleration data has been successfully used for recognition of such activities by various groups, e.g. [Bao and Intille 2004, Kern *et al.* 2003, Krause *et al.* 2003, Van Laerhoven and Gellersen 2004, Lee and Mase 2002, Mantyjarvi *et al.* 2001, Ravi *et al.* 2005].

An important class of activities in healthcare and assisted living are the so-called *Activities of Daily Living* (ADLs). Originally proposed by [Katz *et al.* 1963], they have evolved into a standard set of activities used by physicians and care-givers as a measure to estimate the physical well-being of elderly patients, as well as their need for assisted living. The core set of ADLs consists of the activities *bathing*, *dressing*, *toileting*, *transferring*, *continence*, and *feeding*. The set of ADLs is complemented by the *Instrumental*

Activities of Daily Living (IADLs) proposed by [Lawton and Brody 1969], which consist of *using the phone*, *shopping*, *food preparation*, *housekeeping*, *doing laundry*, *transportation*, *taking medications*, and *handling finances*[1]. Recognition of specific subsets of ADLs and IADLs is demonstrated e.g. by [Philipose *et al.* 2004, Tapia *et al.* 2004, Wang *et al.* 2007, Chen *et al.* 2005, Stikic *et al.* 2008a], who recognize activities such as *making tea, dusting, ironing, vacuuming, cleaning the windows, washing dishes*, or *taking a shower*. Recognizing the complete set of ADLs and IADLs using sensors is challenging, since some activities such as *handling finances* are only loosely defined, and others such as *continence* are difficult to detect. Furthermore, healthcare professionals are often interested not only in the fact that the patient has performed an activity, but also in *how well* the activity was performed. Up to now, automatic estimation of the quality of performing an activity is a largely unsolved research problem.

Apart from the activities mentioned so far, further activities that can be recognized with wearable sensors include sports activities such as *cycling, rowing, running, calisthenics* (e.g. [Ermes *et al.* 2008, Tapia and Intille 2007]) *martial arts moves* [Chambers *et al.* 2002, Kunze *et al.* 2006], *dumbbell exercises* [Minnen *et al.* 2006b], or *juggling* [Huỳnh and Schiele 2006b], furthermore *wood workshop activities* [Lukowicz *et al.* 2004], *assembly tasks* [Ward *et al.* 2006a, Stiefmeier *et al.* 2006, Stiefmeier *et al.* 2008], *reading* [Bulling *et al.* 2008] or *chewing* [Amft *et al.* 2005]. Extremely short-term actvities (also referred to as *gestures*) such as *pulling the handbrake* or *turning a pedal* are explored e.g. by [Zinnen *et al.* 2007, Stiefmeier *et al.* 2007, Benbasat and Paradiso 2001].

2.4.1 High-Level Activities

Interestingly, a large part of research in activity recognition focuses on rather low-level and short-term activities. However, in many applications ranging from healthcare to assisted living to modeling of human behavior, the analysis and recognition of high-level and longer-term activities is an important component. In the following we discuss related work in the area of activity recognition and -discovery, with a focus on authors aiming towards high-level activities. We contrast this work with our own contributions on recognition of high-level activities described in Chapter 6 and Chapter 7.

Let us briefly define the terms *low-level activity* and *high-level activity* as we understand them, since to the best of our knowledge there exists no generally accepted definition of these terms in the activity-recognition community. As *low-level activities* we consider activities such as *walking, sitting, standing, vacuuming, eating, washing dishes*, i.e. activities which can be characterized by (a statistical sequence of) body motion, posture or object use, and which typically last between seconds and several minutes. On an even smaller time scale, brief and distinct body motions such as *taking a step* or *swinging a bat* are sometimes referred to as movements [Bobick 1997], gestures [Ward *et al.* 2005],

[1] As noted by [Philipose *et al.* 2004], the name ADL is commonly used to refer to both ADLs and IADLs in the healthcare community.

or motifs [Minnen *et al.* 2006b]. *High-level activities*, on the other hand, are usually composed of a collection of low-level activities, and are longer-term as e.g. *cleaning the house*, which will typically last more than several minutes and can last as long as a few hours. In this thesis we will sometimes also use the term *scene* to refer to high-level activities. Figure 2.1 illustrates the different activity categories. In the following we report on related work that aims to model and infer what we consider high-level activities.

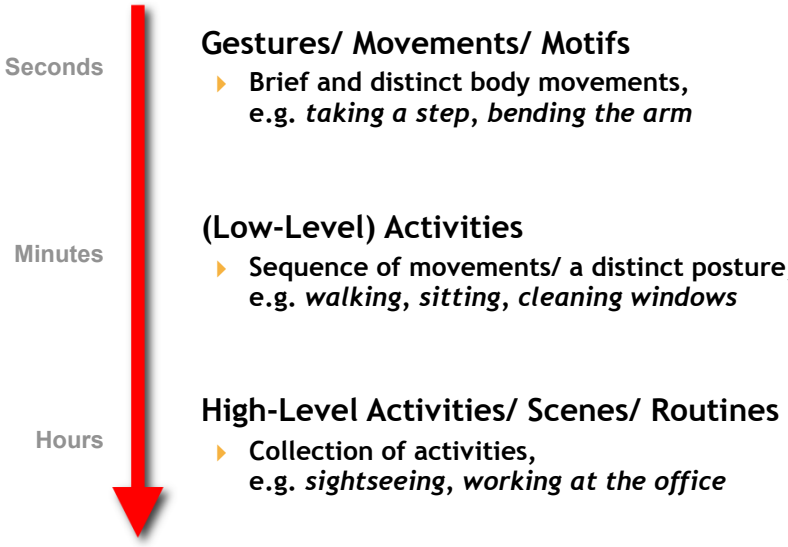


***Figure 2.1**: One of many possible ways to categorize physical activities is to group them based on duration and/or complexity. Note that the terms used for the different categories, and even the categories themselves, vary in the literature.*

Clarkson et al. [Clarkson and Pentland 1999] present an approach for unsupervised decomposition of data from on-body sensors into events and scenes. They use data from wearable sensors to discover short events such as "passing through a door" or "walking down an aisle", and cluster these into high-level scenes such as "visiting the supermarket" by using hierarchies of HMMs. Conceptually, this approach is similar to what our approach described in Chapter 7 can achieve. A notable difference is that our method is able to perform well on low-dimensional, low-resolution data from accelerometers, while the approach in [Clarkson and Pentland 1999] relies on high-dimensional and densely sampled audio and video streams. Since cameras and microphones are often considered intrusive, such an approach will be difficult to put into practice. Thus we believe that the method that we describe in Chapter 7 compares favorably both from a computational and also from a privacy point of view.

Eagle et al. [Eagle and Pentland 2006b] used coarse-grained location and proximity information from mobile phones to detect daily and weekly patterns of location transitions. Their work ultimately focuses on the group rather than the individual and explores themes such as social networks and organizational rhythms.

[Hamid *et al.* 2005] represent activities as bags of n-grams, cluster them into classes of

activities, and characterize these classes by frequently occurring sequences. The patterns they discover on a set of 150 days of a person's indoor location traces are coarse and relatively difficult to interpret, though.

In a more office- and desktop-centered setting, [Oliver *et al.* 2002] use a layered HMM representation to infer office activities such as *giving a presentation*, *having a conversation* or *making a phone call*, based on low-level information from audio and visual sensors as well as from the user's keyboard and mouse activity. In a similar setting, [Horvitz *et al.* 2002] combine device usage with calendar data and time of day/ time of week information to infer a user's availabily. [Begole *et al.* 2003] analyze and visualize daily rhythms of office workers by measuring how active (indicated by computer usage) a person is during different times of day.

There is a significant amount of work that uses location sensors to extract high-level information about a person's activities. The authors in this field often use terms such as *high-level activity* when referring to more meaningful descriptions of low-level position information (such as latitude and longitude), the latter being difficult to interpret by humans. This is slightly different from the view taken in this thesis, in which we consider high-level activities rather as a collection of related low-level activities. An example of work that uses location sensors is given by [Liao *et al.* 2007], who use information from GPS sensors to construct models of high-level activity (such as *work*, *leisure*, *visit*) and to identify significant places. Similarly, [Krumm and Horvitz 2006] use location sensors to make high-level predictions about driving destinations. These works show that location is a powerful cue to the high-level structure of daily life. However, location is often not enough to identify daily routines reliably, as different activities can be performed at the same location. E.g. at home, many people are having dinner and breakfast but also perform work. Similarly, in an office room one might work, hold meetings and even occasionally have lunch. Therefore, we consider the work that we describe in Chapter 7 complementary to these approaches, in that the use of accelerometers allows detection of more fine-grained activities and can also account for different activities performed at the same location.

[Amft *et al.* 2007] introduce a model to detect composite activities composed of atomic events from a variety of body-worn and environmental sensors. In contrast to our work described in Chapter 7, they focus on relatively short sequences, and the method relies on a significant amount of supervision.

Finally, work by Van Laerhoven et. al [Van Laerhoven 2007, Van Laerhoven *et al.* 2008b, Van Laerhoven *et al.* 2008a] explores continuous and long-term monitoring of daily activities with a wrist-worn device, with the goal of developing better techniques for *actigraphy*, a method used in the medical domain e.g. for circadian rhythm analysis, monitoring of wake-sleep patterns, and psychiatric trials with bipolar patients. While their approach is in principle well-suited for recording of high-level activities, it has a different focus, emphasizing on small, light-weight and power-efficient hardware design, as well as efficient online algorithms and compact data representations.

2.5 Learning to Model and Recognize Activities

In the following we give an overview of machine learning methods that have been used for activity recognition with wearable sensors. Our descriptions of the basic machine learning concepts and algorithms will be brief and informal – for a comprehensive mathematical treatment see e.g. [Hastie *et al.* 2001]. As we will see, quite a range of different methods have been employed. A comparison between methods is difficult for several reasons, one being the lack of standard data sets in the area of activity recognition, another the fact that the appropriate approach usually depends on the type of activities and/ or the type of data that has been recorded.

2.5.1 Supervised vs. Unsupervised Approaches

In general one can make the distinction between *supervised* and *unsupervised* learning methods. Supervised learning, sometimes called "learning with a teacher", requires labeled data on which an algorithm is trained before it is able to classify unknown data. By contrast, unsupervised learning, or "learning without a teacher", tries to directly construct models from unlabeled data, either by estimating the properties of its underlying probability density (called *densitity estimation*) or by discovering groups of similar examples (called *clustering*). Until now, supervised learning has been the predominant approach for activity recognition.

Supervised Approaches. The general procedure for training and testing a supervised learning algorithm for activity recognition consists of the following five steps: (1) acquiring sensor data of activities, including annotations of what the user did when (the so-called *ground truth*), (2) transforming the data into application-dependent *features*, e.g. by computing specific properties, eliminating noise, normalizing the data or reducing its dimensionality, (3) dividing the features into a training and a test set, (4) training the algorithm on the training set, and (5) testing the classification performance of the trained algorithm on the test set. Commonly, steps (3) to (5) are repeated with different partitioning into training- and test set, and the results are averaged – this is called *crossvalidation*, and it provides a better estimate of the generalization capability of the algorithm.

There exists a wide range of algorithms and models for supervised learning. Commonly used methods in the context of activity recognition include *Naïve Bayes classifiers* (e.g. [Van Laerhoven *et al.* 2003, Tapia and Intille 2007, Bao and Intille 2004, Maurer *et al.* 2006, Huỳnh and Schiele 2006a]), *C4.5 decision trees* (e.g. [Bao and Intille 2004, Maurer *et al.* 2006, Tapia and Intille 2007]), and *nearest neighbor* methods (e.g. [Kunze *et al.* 2006, Huỳnh *et al.* 2007, Cakmakci and Van Laerhoven 2000, Ravi *et al.* 2005, Bao and Intille 2004]). *Hidden markov models* (HMMs) are well-suited for capturing temporal patterns in the data, but can be difficult to train due to an abundance of parameters (e.g. [Ward *et al.* 2006a, Lester *et al.* 2005, Patterson *et al.* 2005,

Huỳnh *et al.* 2007]). HMMs can be considered as a simple type of *dynamic bayesian network* (DBN), and more complex types of DBNs have also been used [Wang *et al.* 2007, Philipose *et al.* 2004]. Other methods that have been applied include *support vector machines* (SVMs) [Ravi *et al.* 2005, Huỳnh *et al.* 2007, Loosli *et al.* 2003], and more recently *string-matching* methods [Stiefmeier *et al.* 2008, Amft *et al.* 2007]. [Cakmakci and Van Laerhoven 2000] use Kohonen Self-Organising Maps for online-learning of activities such as *walking, running* and *riding a bike*. *Boosting* is a so-called meta-classifier, in which a collection of weak classifiers with accuracies just above chance level is combined into a single and possibly very accurate classifier (e.g. [Wang *et al.* 2007, Stikic *et al.* 2008a, Minnen *et al.* 2007, Ravi *et al.* 2005]). As noted in [Van Laerhoven 2005], boosting methods are well-suited for online implementations in distributed sensor networks, in which each sensor node can assume the role and the processing tasks of a weak learner.

Unsupervised Approaches. An unsupervised learning procedure typically consists of (1) acquiring unlabeled sensor data (2) transforming the data into features, and (3) modeling the data using some kind of density estimation or clustering. During clustering, for instance, each data point is assigned to one (or more) of N groups of points that are close with respect to a predefined distance measure. Evaluation of unsupervised approaches is usually difficult due to the lack of ground truth to which one can compare the discovered structure. In our studies on unsupervised learning of activities described in Chapter 4, Chapter 5 and Chapter 7, ground truth was available which we used to evaluate our approach in a similar fashion as for the supervised case.

[Clarkson and Pentland 1999] use hierarchies of HMMs to learn locations and scenes such as *walking through the supermarket* from audio and video data in an unsupervised fashion. [Liao *et al.* 2007, Patterson *et al.* 2004] use unsupervised learning schemes based on graphical models. Their focus is on inferring transportation modes (such as *bus, car, walking*) and destination goals of the user. [Minnen *et al.* 2006b] combine discrete string matching techniques with continuous HMM classifiers to discover short recurring motifs in acceleration data. They aim to discover and model short term motion primitives, such as those occurring during physical exercise. [Huỳnh and Schiele 2006b] use the concept of multiple eigenspaces for unsupervised learning of activities such as *walking* or *juggling*.

Semi-Supervised Approaches. *Semi-supervised* learning methods represent a third class of methods that can be applied when parts of the available data are labeled, while for other (possibly large) parts there exist no labels. Semi-supervised learning is appealing for activity recognition, where it is usually expensive to obtain continuous ground truth, but feasible to ask the user to label small parts of the recordings. Until now there has been relatively little work on semi-supervised learning for activity recognition with wearable sensors [Subramanya *et al.* 2006, Stikic *et al.* 2008b]. A semi-supervised approach for activity recognition is described in Chapter 5 of this thesis [Huỳnh and Schiele 2006a].

2.5.2 Discriminative vs. Generative Models

State-of-the-art activity recognition algorithms can roughly be divided in two groups concerning the choice of the classifier, one group using generative models and the other discriminative models. Generative approaches infer the class-conditional distributions $p(\mathbf{x}|C_i)$ of the input data $\mathbf{x}$ given class C_i. Sampling from these distributions allows the creation of new data points in the input space, hence the name generative models. Together with an estimate of the prior class probabilities $p(C_i)$, the posterior class probabilities $p(C_i|\mathbf{x})$ can be determined via Bayes' theorem

$$p(C_i|\mathbf{x}) = \frac{p(\mathbf{x}|C_i)p(C_i)}{p(\mathbf{x})}. \tag{2.1}$$

The normalizing factor $p(\mathbf{x})$ in equation 2.1 can be determined by

$$p(\mathbf{x}) = \sum_i p(\mathbf{x}|C_i)p(C_i) \tag{2.2}$$

Discriminative approaches, on the other hand, try to directly solve the problem of determining the posterior class probabilities $p(C_i|\mathbf{x})$, without modeling the class-conditional densities $p(\mathbf{x}|C_i)$. Thus they focus on learning the class decision boundaries rather than modeling the properties of the individual classes.

Generative models are appealing for several reasons in the context of activity recognition. For example, these models can be learned incrementally or even in a fully unsupervised fashion (as shown for example in Chapter 4 [Huỳnh and Schiele 2006b]), they can deal with missing data in a principled way, they allow for modular construction of composed solutions to complex problems and therefore lend themselves to hierarchical classifier design. Also, prior knowledge can be easily taken into account. However, the price for these favorable properties is that generative models tend to produce a significant number of false positives. This is particularly true for activities that are rather similar such as *walking* and *walking upstairs*. Therefore it is difficult to scale these approaches to a wide range of sometimes highly similar activities.

Discriminative methods enable the construction of flexible decision boundaries, resulting in classification performances often superior to those obtained by purely probabilistic or generative models [Jaakkola and Haussler 1998, Ng and Jordan 2002]. Related work in the computer vision community has shown that this allows for example to explicitly learn the discriminant features of one particular activity or between multiple activities [Torralba *et al.* 2004, Nilsback and Caputo 2004]. Also, recent work has shown the suitability of discriminative methods for recognition of large numbers of activities [Torralba *et al.* 2004].

There has been an increasing interest in the machine learning community in developing algorithms which combine the advantages of discriminative methods with those

of probabilistic generative models [Jaakkola and Haussler 1998, Lasserre *et al.* 2006], showing improvements in performance with respect to purely discriminative or generative approaches in information extraction (e.g. from biomedical text and gene finding [Perreira *et al.* 2004]). However, the activity recognition community has so far typically chosen one of these two modeling approaches (with the notable exception of [Lester *et al.* 2005]). In Chapter 5 [Huỳnh and Schiele 2006a] we introduce a combined method which leverages advantages of both approaches.

3

Features for Activity Recognition

In many wearable computing scenarios, basic activities such as *walking*, *standing* and *sitting* are inferred from data provided by body-worn acceleration sensors. In such settings, most existing approaches employ a single set of features, regardless of the activity to be recognized. In this chapter we show that recognition rates can be improved by careful selection of individual features for each activity. We present a systematic analysis of features computed from a real-world data set and show how the choice of feature and the window length over which the feature is computed affects the recognition rates for different activities. Finally, we give a recommendation of suitable features and window lengths for a set of common activities.

3.1 Introduction

In this chapter we focus on finding suitable features for activity recognition tasks. As *features* we consider the result of the transformation of raw sensor data into another space – the *feature space* – in which the classification task can be solved more easily. The choice of features strongly influences the result of the final classification and therefore is an important step in the design of any activity recognition system. A typical approach in activity recognition is to compute the features locally over a *sliding window* which is shifted over the stream of sensor data. This introduces two additional parameters, namely the length of the window and the amount of shift between consecutive windows.

Comparing the different approaches to activity recognition, one can observe that a common approach is to decide on a fixed set of features and a fixed window length and use this combination for the whole set of activities to be recognized. Even though the resulting recognition rates can be generally high, they might be improved by selecting features and window lengths for each activity separately.

In the following we propose to use cluster analysis to rank individual features with respect to a given activity. We then show that the ranking obtained from the cluster analysis directly translates to recognition results, and is therefore a valid measure for the quality

of a feature with respect to a given activity. We then use this measure to evaluate different features and window lengths on a data set of six different activities.

The rest of the chapter is organized as follows. Section 3.2 introduces the data set that we used for our work. In Section 3.3, we give an overview of commonly used features from accelerometer data, and then describe the features that we used and how we analyzed them using cluster analysis. We report on the results of the analysis and discuss the influence of different features and window lengths. In Section 3.4, we show that there is a direct correspondence between cluster precision and recognition rate for a given activity, and illustrate this with several examples. Finally, in Section 3.5 we summarize our results and draw conclusions.

3.2 Dataset

For our experiments, we used data recorded by Intel Research, Seattle [Lester *et al.* 2005]. The subset we used consists of roughly 200 minutes of sensor data recorded by two subjects who were not affiliated with the researchers. The subjects were given a script containing the activities to perform, namely *walking*, *standing*, *jogging*, *skipping*, *hopping* and *riding bus* (the latter consisting mostly of *sitting*). They recorded these activities in everyday life situations without supervision of a researcher. Later the data was annotated with the help of recorded video and audio data.

The data was recorded using an integrated sensor board developed by Intel Research that was attached to the shoulder strap of a backpack the subjects were carrying. The board contains sensors for 3D-acceleration, audio, temperature, IR/visible/high-frequency light, humidity and barometric pressure, as well as a digital compass.

3.3 Feature Analysis

Our goal for this study was to identify suitable features and window lengths for recognition of common activities. We did not want to commit ourselves to a particular recognition algorithm, as this would have limited the degree to which the results can be generalized. Therefore, we decided to use a different measure based on cluster analysis, similar to the one proposed by [Mikolajczyk *et al.* 2005], which we introduce in the following.

Clustering uncovers structure in data by grouping it according to a given distance measure. Our rationale for this study was that if clusters of activity data were homogeneous in terms of their labels, a recognition algorithm would be able to differentiate well between the activities. Thus, in this section we propose a simple measure of cluster homogeneity and use it to rank features w.r.t. activities. We then show that this measure is indicative of recognition performance by feeding the features to a simple classifier. We use a simple classifier because the purpose of the classifier is not to yield high recognition rates,

but to show that the results of the cluster analysis can be used to decide on features for recognition.

As we were mainly interested in evaluating the performance of individual features for activity recognition, we confined ourselves to one-dimensional features, both for the clustering and the subsequent recognition. Using the knowledge gained from the feature analysis, one can later easily combine them to form higher-dimensional features and/or use them in a more elaborate classifier scheme such as the popular HMM or SVM frameworks.

3.3.1 Commonly used Features from Accelerometers

Popular features computed from the acceleration signal are mean [Bao and Intille 2004, Kern *et al.* 2003, Heinz *et al.* 2003, Krause *et al.* 2003, Ravi *et al.* 2005], variance or standard deviation [Kern *et al.* 2003, Heinz *et al.* 2003, Lee and Mase 2002, Ravi *et al.* 2005], energy [Bao and Intille 2004, Ravi *et al.* 2005], entropy [Bao and Intille 2004], correlation between axes [Bao and Intille 2004, Ravi *et al.* 2005] or discrete FFT coefficients [Krause *et al.* 2003]. Energy and entropy are usually derived from the latter. [Van Laerhoven and Gellersen 2004] use peaks in raw data; [Mantyjarvi *et al.* 2001] use powers of wavelet coefficients. The window length over which the features are computed is usually fixed, e.g. 6.7 sec in [Bao and Intille 2004], 1 sec in [Kern *et al.* 2003], 2 sec in [Mantyjarvi *et al.* 2001], 8 sec in [Krause *et al.* 2003] and 5.12 sec in [Ravi *et al.* 2005].

3.3.2 Features used in this Study

This work focuses on features derived from accelerometers, as these have been successfully used for the activities we are considering. We decided on a set of features and window lengths that have been commonly used in related work (see Section 3.3.1 above). The features were computed over windows of 128, 256, 512, 1024 and 2048 samples. At a sampling rate of 512 Hz, the lengths correspond to 0.25, 0.5, 1, 2 and 4 seconds, respectively. The windows were shifted over the data in steps of 0.25 seconds. From each window, we compute mean, variance, energy, spectral entropy, as well as discrete FFT coefficients. The FFT coefficients were grouped in six exponential bands, and another 19 features were obtained by pairwise addition of coefficients 1+2, 2+3, ..., to 19+20. In addition to that, we computed three features representing the pairwise correlation of the x-, y- and z-axis. Apart from the features computed from the accelerometer data, we included the variance of the digital compass and of the visible light sensor.

3.3.3 Clustering

Ideally, when clustering data in feature space, each cluster should contain samples of only one activity. This would indicate that the data of the given feature was clearly separable

and thus well-suited as an input for classification. In the worst case, the fraction of samples of an activity in each cluster would be equal to the a priori probability of the activity. This would imply that the feature was not discriminative for the given set of activities and thus unlikely to be suited for recognition.

While there exist a number of measures for the quality of a clustering, such as purity or normalized mutual information (see e.g. [Manning *et al.* 2008]), these measures usually provide a single average result over all classes. As we are interested in the distribution of individual activities in the clusters, we used a measure that reflects these distributions better. We describe this measure in the following.

In order to measure the distribution of samples for different activities in the clusters, we first compute for each cluster i and activity j the fraction

$$p_{ij} = \frac{|C_{ij}|}{\sum_j |C_{ij}|} \tag{3.1}$$

where $|C_{ij}|$ is the amount of samples in cluster i labeled with activity j. We then form a weighted sum of these fractions to obtain a cluster precision p_j for each activity j:

$$p_j = \frac{\sum_i |C_{ij}| p_{ij}}{\sum_i |C_{ij}|} \tag{3.2}$$

Thus, if an activity's cluster precision is close to one, this indicates that there are many clusters mainly consisting of samples for this activity. We weight each fraction by the number of samples it represents in order to prevent smaller clusters from dominating the result.

In order to reduce overfitting artifacts, we did not directly measure cluster precision on the result of the clustering, but instead applied a five-fold cross validation scheme as follows. The data was first randomized and divided into five equally sized partitions. K-means clustering was then applied to four of the five partitions, the fifth being left for testing. Testing was done by assigning each sample in the test partition to the nearest cluster, based on distance from sample to cluster centroid. The cluster precision was then measured on the assignments of the test partition.

3.3.4 Results

In the following we report on the results of the feature analysis. We first discuss the influence of the different features, and then of the different window lengths over which the features were computed.

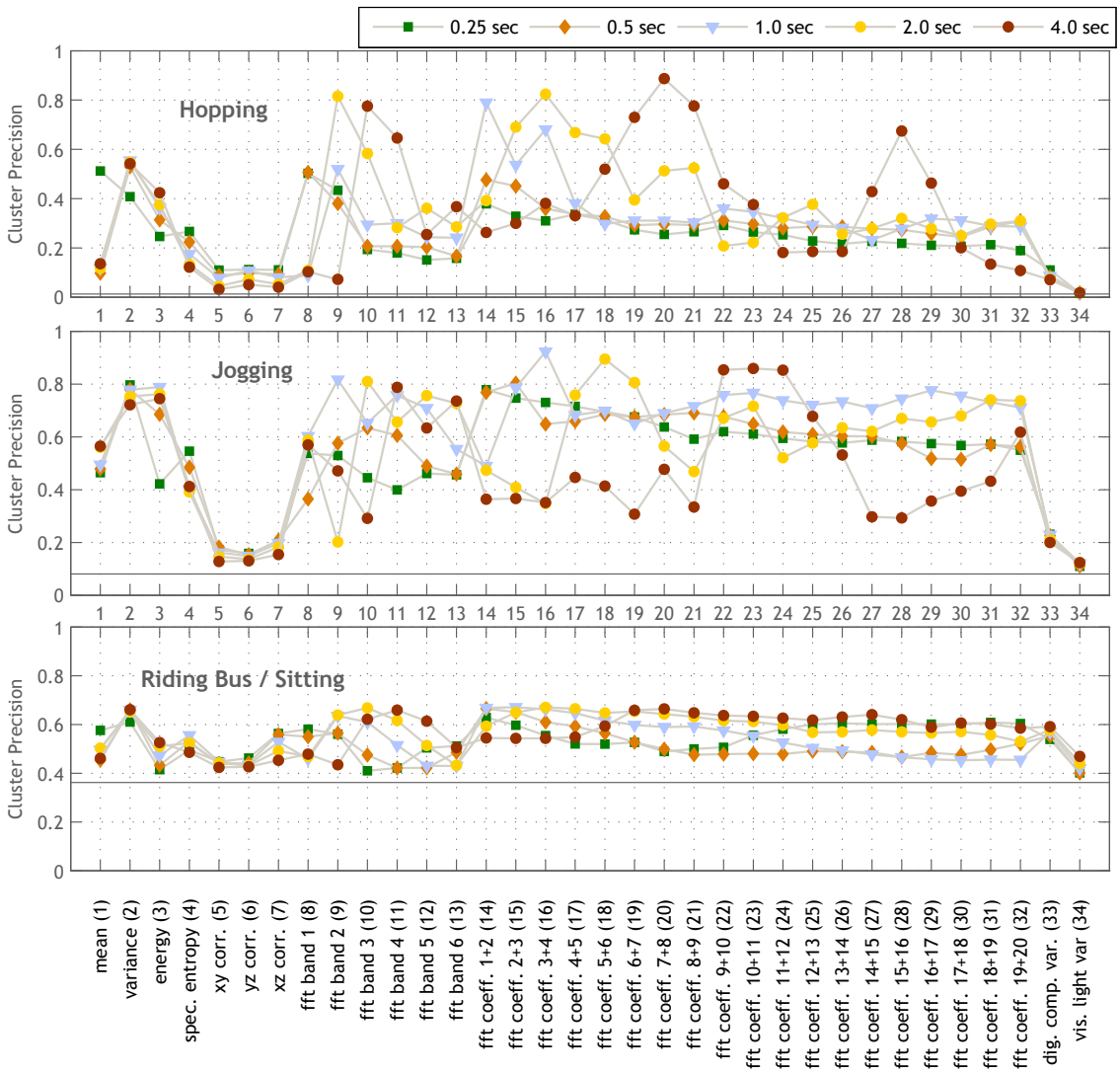


***Figure 3.1**: Cluster Precision for the activities* hopping, jogging *and* riding bus (sitting), *for different features and window lengths. The horizontal line in each plot marks the a priori probability of the activity.*

The feature computation yielded about 50,000 samples for each type of feature [1].The results we show are based on $k = 1000$ clusters (we also evaluated other numbers of clusters (e.g., 100), but the results vary relatively little). Figures 3.1 and 3.2 show the average cluster precision per feature and window length (a more condensed summary view is shown in Figure 3.7). Each plot corresponds to one activity, and each line in a plot represents one window length. Note that the lines connecting the different values are only drawn for better readability, and have no meaning apart from that. The horizontal line in each plot marks the a priori probability of the activity.

[1]Since we used the same amount of shift for all window lengths, the actual window length had little influence on the total number of samples.

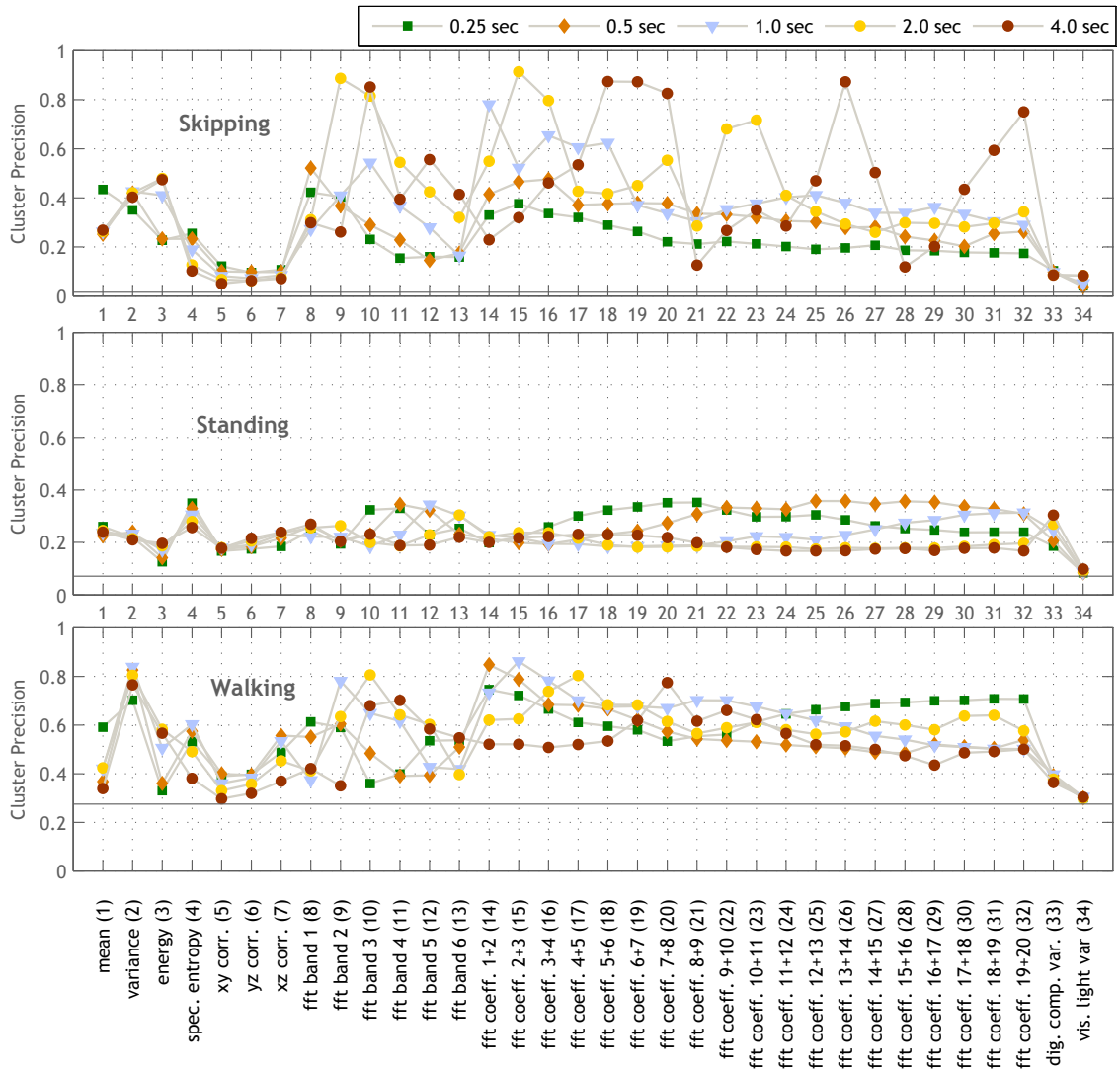


***Figure 3.2**: Cluster Precision for the activities* skipping, standing *and* walking, *for different features and window lengths. The horizontal line in each plot marks the a priori probability of the activity.*

The plots in Figures 3.1 and 3.2 indicate a clear difference between the cluster precision of stationary activities such as *standing* and *riding bus* (the latter consisting mainly of *sitting* in the bus) and the moderate to high intensity activities, namely *walking*, *jogging*, *hopping* and *skipping*. One can observe that the variance in the cluster precision of different features is much higher for activities with moderate to high intensity levels. Not surprisingly, for these activities the FFT features perform clearly better than most of the other features. However, there is much variation between the cluster precision of the different FFT coefficients. E.g., for *skipping*, when using a window length of 4 seconds, the cluster precision between FFT coefficients 13+14 and 15+16 drops from 0.9 by almost 80% to 0.12. Similar differences can be observed for *hopping* and *jogging*. Furthermore,

no single FFT feature is best for all activities. The coefficients 1+2 are among the top five features for *walking*, *hopping* and *riding bus*. Coefficients 2+3 have the highest precision of all features for *skipping*, *walking* and *riding bus*. Coefficients 3+4 attain high precision for *jogging*, *hopping* and *riding bus*. For all other coefficients, no clear statement across multiple activities can be made. Instead, one has to take a close look at each activity to see which coefficients are best. For *standing*, coefficients 12 to 16 and 7 to 8 perform best, for *jogging* coefficients 3+4, and for *hopping* coefficients 7+8. For *walking* and *riding bus* variance does remarkably well, being in third and fourth place, respectively. For *walking*, *riding bus* and *hopping*, the third exponential FFT band might serve as a compromise to the FFT coefficients, since it ranks among the first five features for these activities.

Comparing the different window lengths to each other, we observe that for *walking*, *jogging* and *riding bus*, the 1 second window attains the highest precisions on average. For *skipping* and *hopping*, the 2 and 4 second windows score best on average, while the 0.25 and 0.5 second windows attain relatively low precision for all features of these two activities. For *standing*, the short windows of 0.5 and 0.25 seconds achieve high precision for a range of FFT coefficients. The longer windows of 2 and 4 seconds are not suited for *standing* – the precision for these window lengths is quite low. In contrast to this, *jogging* has some peaks with more than 80% precision for 1, 2 and 4 second windows. The 0.25 and 0.5 second windows work not very well for *jogging*, except for the FFT coefficients 1+2.

When looking at features and window lengths combined, the following are the best combinations per activity: *hopping*: FFT coeff. 7+8 over 4 sec; *skipping*: FFT coeff. 2+3 over 2 sec; *jogging*: FFT coeff. 3+4 over 1 sec; *riding bus/sitting*: FFT coeff. 2+3 over 1.0 sec; *walking*: FFT coeff. 2+3/ 1 sec; *standing*: FFT coeff. 12+13/0.5 sec. It should be noted that for most activities, there is more than one combination that performs well, as can be seen from Figures 3.1 and 3.2.

In conclusion, an important result of our analysis is that there are features and window lengths which perform well across different activities, but in order to achieve best performance one should choose features separately for each activity.

3.4 Recognition

In this section we show that our measure of cluster precision can serve as an indicator of recognition performance, by comparing our results to the output of a simple classifier. We construct the classifier by dividing the computed features into training and test sets in the same fashion as the cluster analyis, then apply k-means clustering on the training set and label each training cluster i with its dominating activity $\hat{j} = \text{argmax}_j(p_{ij})$. Each sample of the test set is then either classified according to the label of the nearest centroid i, if $p_{i\hat{j}} > t$ for a given threshold t, or as *unknown* otherwise. Varying the threshold t between 0 and 1 allows us to plot a precision-recall curve for a given activity and feature.

3.4.1 Results

To test our hypothesis that cluster precision is an indicator of recognition performance, we chose the activity *walking* and used the three features with the highest cluster precision as input for the classifier. The results are shown in Figures 3.3, 3.4 and 3.5. One can observe that the order imposed on the curves by the equal error rates (the intersections of the curves with the diagonal line) and by the precision p (shown in square brackets) are the same for all three plots. This indicates that for a given feature, we can use the cluster precision of different window lengths to estimate how well recognition rates for a particular window length will be.

In order to be useful, the results of the cluster analysis must not only generalize across different window lengths, but also across different features. Next, we validate this by comparing the recognition rates of different features to each other.

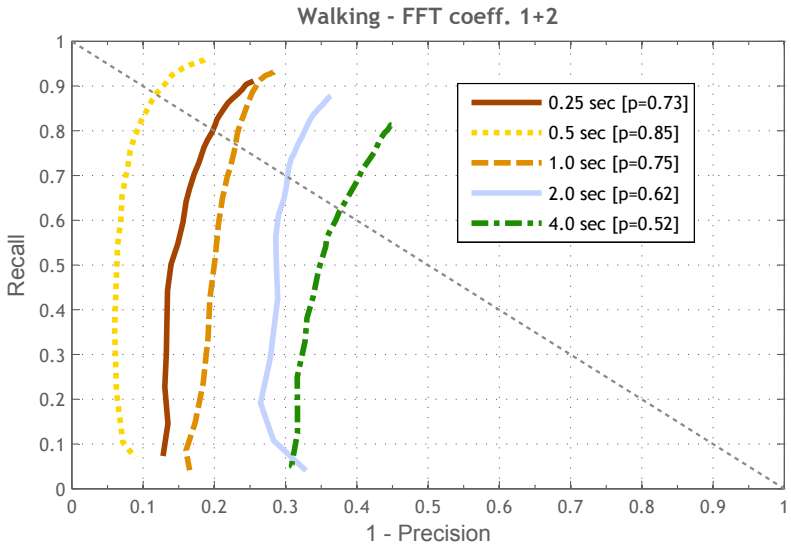


***Figure 3.3**: Recognition results for the activity* walking *using the FFT coefficients 1+2 computed over different window lengths.*

Figure 3.6 shows the recognition results for the six different activities, using the five best combinations of feature and window length in terms of cluster precision p (shown in square brackets in the legend). In most cases these are FFT coefficients. Recognition of *jogging* and *walking* performs particularly well, with equal error rates up to about 90%. Note that many curves are very steep, indicating that by lowering the threshold of the classifier, higher recall can be obtained without sacrificing precision.

Our main goal, however, was less to attain high recognition rates than to investigate to what extent the results of the cluster analysis generalize to the recognition results. One can see that in most cases, the order is preserved, i.e. features with higher cluster precision also have better recognition rates. For *standing* and *riding bus* there are only small differences in the equal error rates, just like in the precision values. For *skipping*,

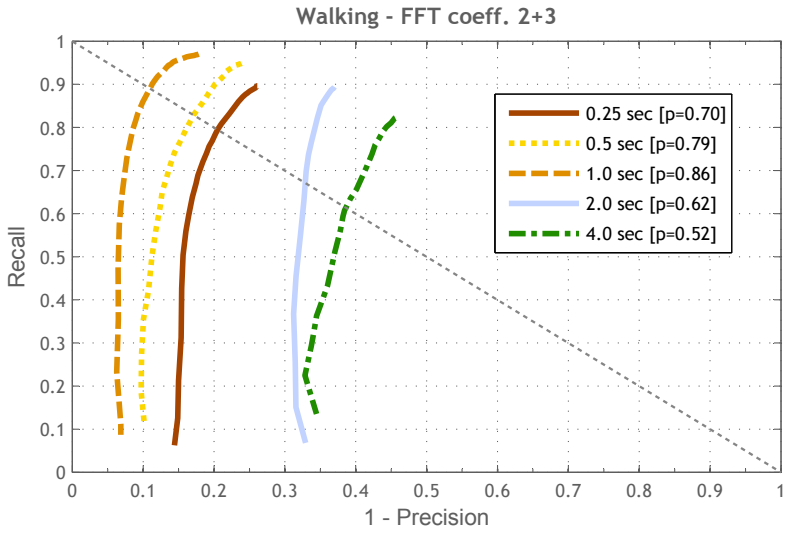


***Figure 3.4**: Recognition results for the activity* walking *using the FFT coefficients 2+3 computed over different window lengths.*

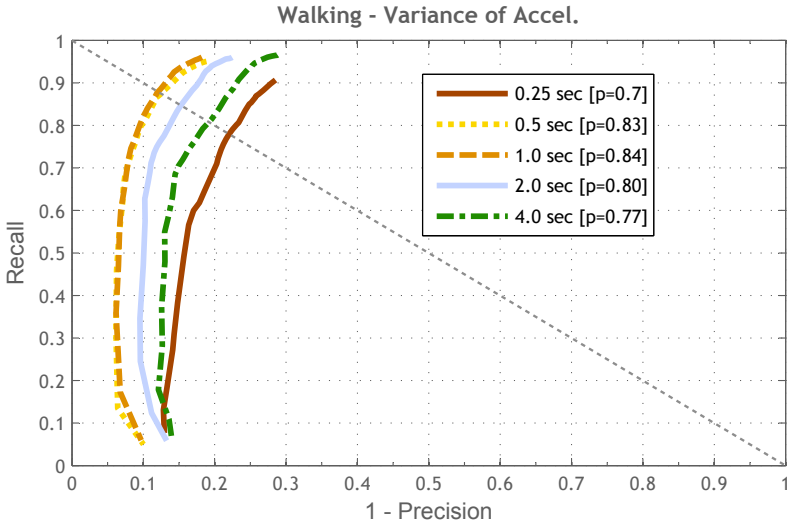


***Figure 3.5**: Recognition results for the activity* walking *using the the variance of the acceleration signal as feature.*

the order is preserved except for one feature (FFT coefficients 2+3). A possible reason for this is that the differences in cluster precision for these features are very small. Also, the samples for *skipping* constitute only about 1.5% of the total number of samples, which might introduce artifacts.

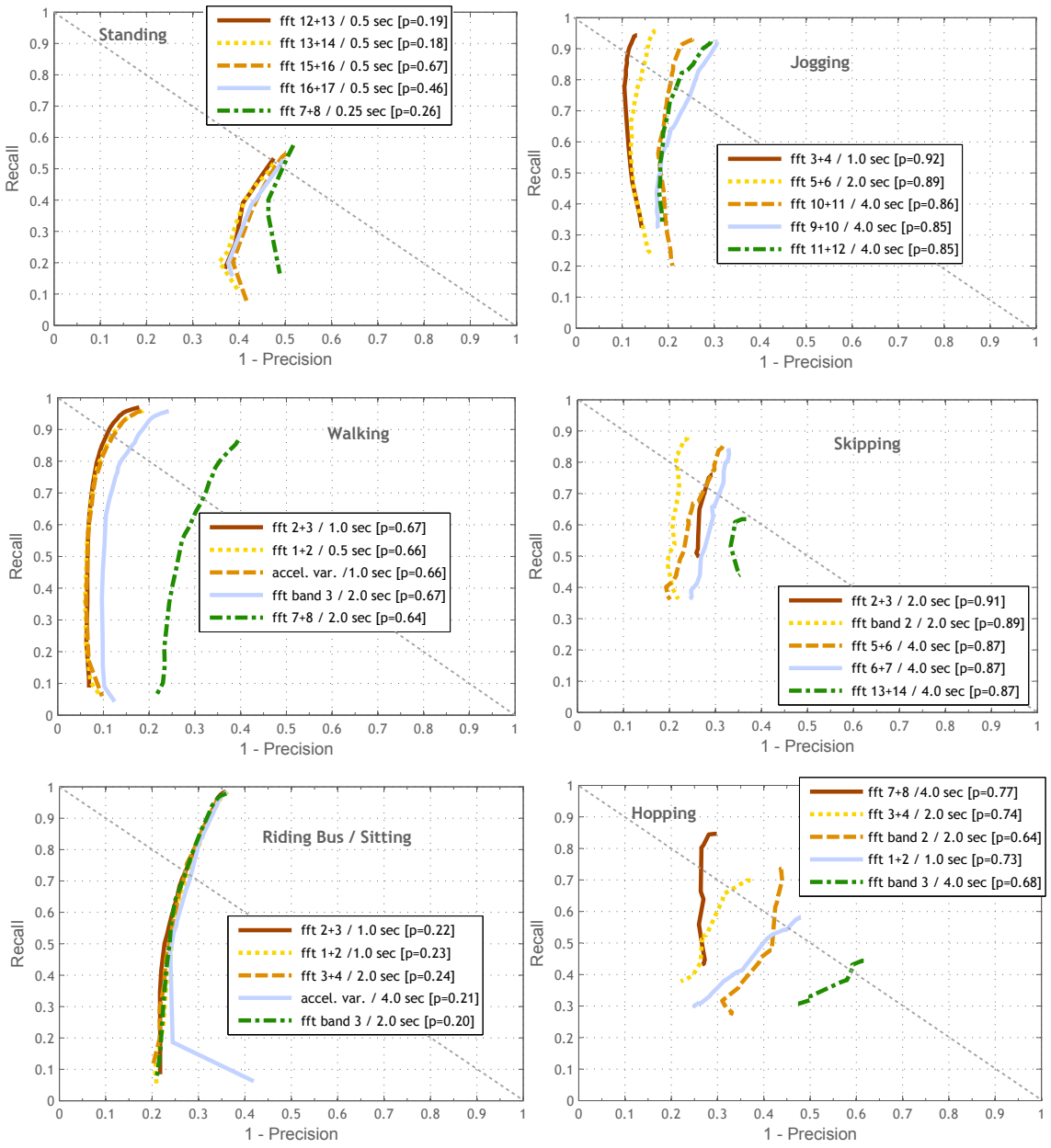


Figure 3.6: *Recognition results for different activities. For each activity, the five best combinations of features and window lengths in terms of cluster precision are shown. For each combination the cluster precision p is indicated in square brackets.*

3.5 Summary and Conclusion

We have seen that by clustering features and by comparing them to each other in terms of cluster precision, one can obtain detailed information about how well a particular feature is suited for activity recognition. Our proposed measure of cluster precision turned out to be a good indicator for the recognition performance of a feature. We gave a detailed

comparison of the cluster precisions of a range of features and showed that the ranking obtained from the cluster analysis is reflected in the recognition rates of the different features.

Overall, our results indicate that in contrast to an assumption that is sometimes implicitly made, there is neither a single feature nor a single window length that will perform best across all activities. Figure 3.7 shows a summary of the results w.r.t. to the different features, averaged over all window lengths. By looking at the different features, we found that the FFT features always rank among the features with the highest cluster precision. However, the FFT coefficients that attain the highest precision are different for each activity, and recognition can be improved by selecting features for each activity separately. Our recognition results also indicate that combining different FFT coefficients to bands of exponentially increasing size might be a compromise to using individual or paired coefficients. For the non-FFT features, we found that variance often performs well. Surprisingly, the often-used mean of the acceleration signal has lower precision values than variance throughout the set of activities, except when used with a window length of 0.25 seconds for *jogging* and *skipping*.

In terms of window lengths, we found that on average, features computed over window lengths of one and two seconds attain slightly higher precision values than those computed over other window lengths. However, there are significant differences across the different activities, and as for the features, selecting different window lengths for different activities leads to better recognition rates. E.g., the 1 second window has the highest average precision values for the activities *jogging* and *walking*; the 2 and 4 second windows attain high values for *skipping* and *hopping*, and the 0.25 and 0.5 second windows reach relatively high precision for the activity *standing*.

Our results indicate which features and window lengths are good candidates for recognizing certain activities. For specific scenarios, however, the choice of feature can depend on additional factors such as the nature of the dataset, the type, characteristics, and position of the sensors used for recording, or on computational constraints. For instance, the computation of frequency features is relatively expensive in terms of processing power, so that one might decide in favor of simpler features when aiming for devices with limited processing capabilities. These considerations should be kept in mind when applying our results.

In the next chapter we will turn to the first of the two challenges that are the main focus of this thesis, namely the development of methods for activity recognition which require little or no supervision during training.

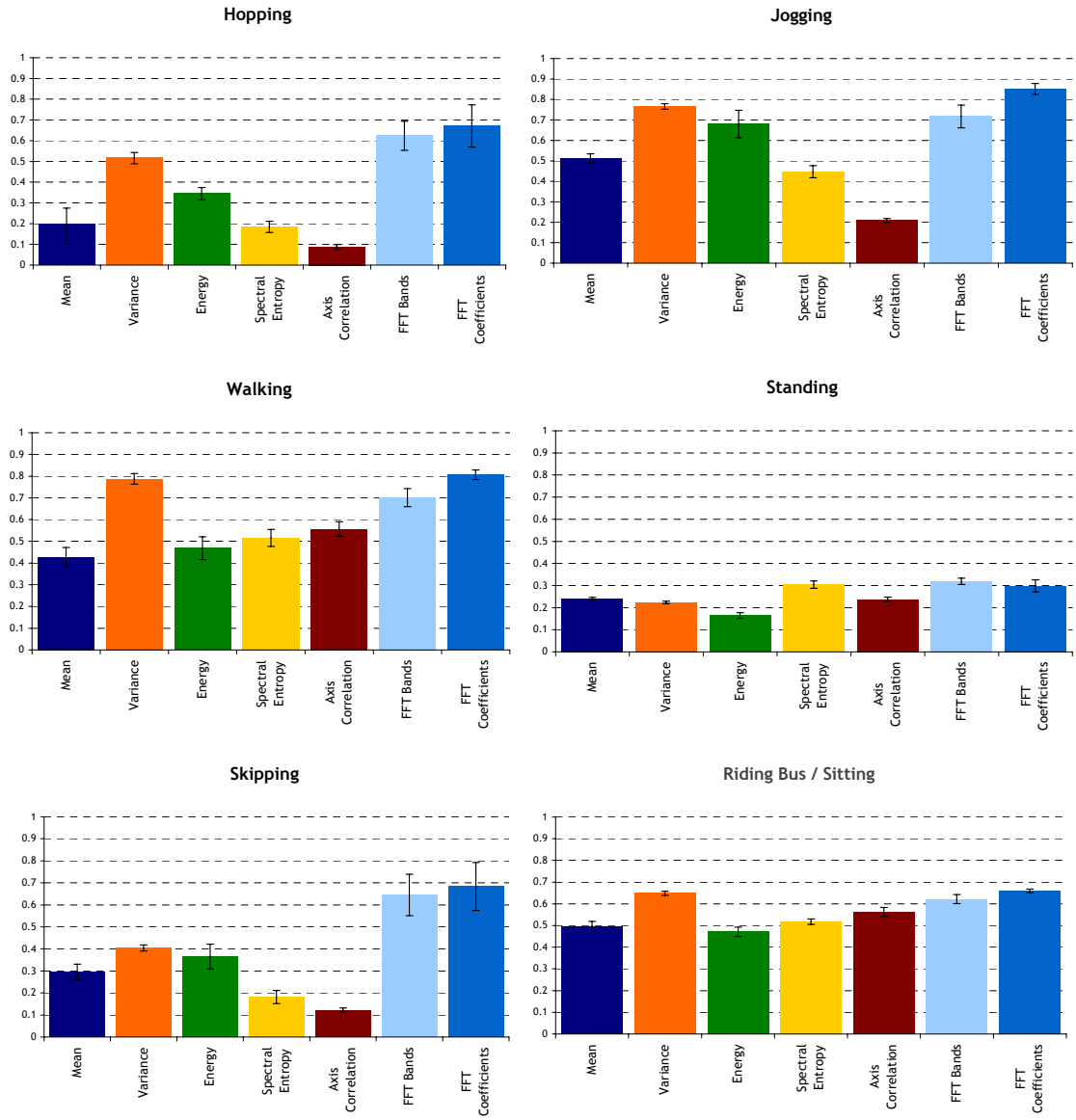


***Figure 3.7**: Summary view: for each activity, the performance of the different features in terms of cluster precision is shown. The results are averaged over all window lengths.*

4

Unsupervised Learning of Activities

In this chapter we propose a novel scheme for unsupervised detection of structure in activity data. Our method is based on an algorithm that represents data in terms of multiple low-dimensional eigenspaces. We describe the algorithm and propose an extension that allows to handle multiple time scales. The validity of the approach is demonstrated on several data sets and using two types of acceleration features. Finally, we report on experiments that indicate that our approach can yield recognition rates comparable to current supervised approaches.

4.1 Introduction

Activity recognition is typically based on supervised learning techniques. As we have already argued in Section 1.1.1, it is desirable to reduce the amount of supervision to a minimum for various reasons. An important argument in favor of less supervision is that for large amounts of data and/ or many activity classes, labeling simply becomes impractical and error-prone. Also, in order to quickly adapt to different users and usage scenarios, a context aware system should be able to support adaption through unsupervised learning techniques with minimal feedback.

Motivated by these considerations, we will next propose an unsupervised approach to discover structure in sensor data in order to model and recognize human activities. The proposed approach is neither limited to activity learning and recognition, nor to a particular type of sensor. Rather it can be applied more generally to many types of sensors and context information.

The chapter is organized as follows. First, an unsupervised learning scheme for the discovery of activities in sensor data is proposed, based on the concept of multiple eigenspaces. Second, the multiple eigenspace algorithm is extended to handle multiple time scales of sensor data, thereby reducing its dependency on fixed time scales, which are often not known beforehand. Third, an experimental comparison of two different feature representations for the discovery of activities at different time scales are evaluated. Fourth, the algorithm is evaluated on real-world data from body-worn sensors, yielding comparable performance to fully supervised learning approaches.

4.2 Multiple Eigenspaces

Principal component analysis (PCA) is a standard technique in pattern recognition and machine learning to reduce the dimensionality of feature spaces. PCA finds the principal components (or *eigenvectors*) of a data distribution spanning a linear subspace (or *eigenspace*) of the feature space. PCA is an unsupervised technique in the sense that it finds the optimal linear subspace to represent the data without any annotation or user intervention. In many applications however, it is more appropriate to represent the inherent structure of a data-set not with a single but with multiple eigenspaces [Leonardis *et al.* 2002].

In the following we show that the concept of multiple eigenspaces can be used to detect and represent structure such as individual activities in accelerometer data. We first give a formal problem description (Section 4.2.1) and then describe an algorithm to extract multiple eigenspaces (Sections 4.2.2 to 4.2.5). We then propose an extension to the algorithm that can handle multiple time scales (Section 4.2.6). Section 4.3 then gives an example illustrating the different stages of the algorithm.

4.2.1 Problem Description

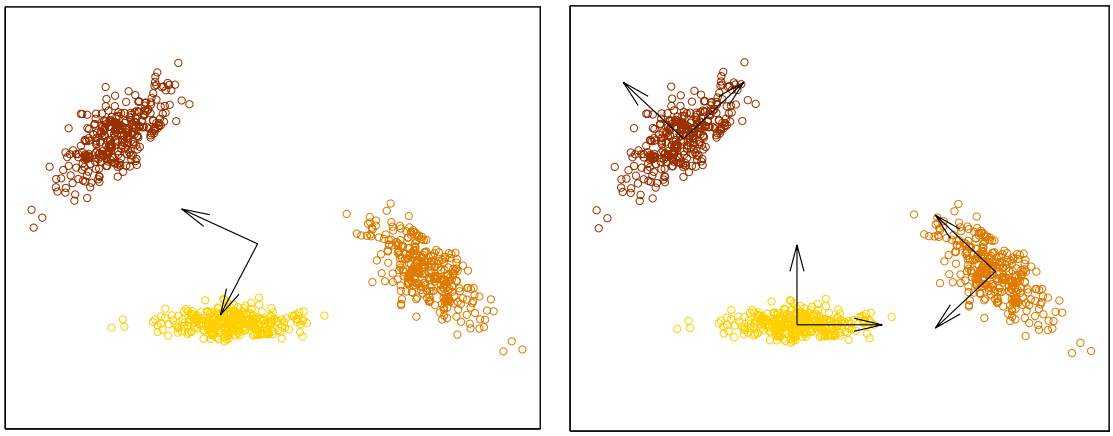
Principle component analysis (PCA) allows to approximate each vector $\mathbf{x}_i$ of a set $\mathcal{G} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m | \mathbf{x}_i \in \mathbb{R}^n\}$ by an approximation $\hat{\mathbf{x}}_i, i = 1, \ldots, m$, so that

$$\hat{\mathbf{x}}_i \quad = \quad \mathbf{e}_0 + \sum_{k=1}^{p} y_k \mathbf{e}_k, \tag{4.1}$$

i.e., by a vector $\mathbf{e}_0 \in \mathbb{R}^n$ plus a linear combination of p (eigen-)vectors $\mathbf{e}_1, \ldots, \mathbf{e}_p$ ($p < n, \mathbf{e}_k \in \mathbb{R}^n$). PCA is optimal in the sense that for a given number of eigenvectors p, the reconstruction error $\varepsilon^2 = \sum_{i=1}^{m} ||\mathbf{x}_i - \hat{\mathbf{x}}_i||$ is minimal.This is achieved by defining $\mathbf{e}_0$ as the mean of all $\mathbf{x}_i \in \mathcal{G}$ and $\mathbf{e}_1, \ldots, \mathbf{e}_p$ as the eigenvectors corresponding to the p largest eigenvalues of the covariance matrix of the vectors in $\mathcal{G}$. We call the linear subspace spanned by $\mathbf{e}_0, \mathbf{e}_1, \ldots, \mathbf{e}_p$ the *eigenspace* of $\mathcal{G}$ of dimension p, in short $\mathcal{E}(\mathcal{G})$. If $p = 0$, then $\mathcal{E}(\mathcal{G})$ only consists of the mean.

If the vectors in $\mathcal{G}$ are sufficiently correlated p can be chosen to be much smaller than the dimension of the original vector space, while still maintaining a low reconstruction error ε^2. In such cases, $\mathcal{E}(\mathcal{G})$, together with the coefficients $y_1, \ldots, y_p$ (see eq. 4.1) of each $\mathbf{x}_i \in \mathcal{G}$, can serve as a low-dimensional representation of $\mathcal{G}$.

In many cases, however, a single linear eigenspace will be too general to capture the low-dimensional structure of the data. Consequently, the dimension of $\mathcal{E}(\mathcal{G})$ must be high in order to obtain acceptable reconstruction errors. Apart from the computational



(a) (b)

***Figure 4.1**: Using multiple instead of a single eigenspace to model a dataset can lead to more compact and low-dimensional representations. In this simple example, the dataset consists of three distinct clusters, so that a single eigenspace will not be able to capture the structure well (a). When using one eigenspace per subset (b), the structure of the data is captured much better, and reducing the dimension of the eigenspaces will lead to smaller reconstruction errors than in (a).*

issues involved, this means that the eigenspace cannot serve as a good representation of the inherent structure of the data. In such cases, it would be more suitable to divide $\mathcal{G}$ into sufficiently correlated subsets $\mathcal{G}_j \subset \mathcal{G}$ and represent those subsets with eigenspaces $\mathcal{E}_j(\mathcal{G}_j)$, or short $\mathcal{E}_j$. Each of those eigenspaces could then serve as a compact and low-dimensional model of the corresponding part of the data. Figure 4.1 illustrates this idea.

The problem to be solved is thus, given a set of data vectors $\mathcal{G}$, to find sets $\mathcal{G}_j \subset \mathcal{G}$, eigenspaces $\mathcal{E}_j$ and dimensions p_j, so that each $\mathbf{x}_i \in \mathcal{G}_j$ can be approximated to a predefined degree of accuracy by its projection

$$\hat{\mathbf{x}}_i \quad = \quad \mathbf{e}_{0j} + \sum_{k=1}^{p_j} y_{kj}\mathbf{e}_{kj}. \tag{4.2}$$

4.2.2 Overview of the Multiple Eigenspace Algorithm

Leonardis et al. [Leonardis *et al.* 2002] proposed an iterative procedure to solve the above problem by simultaneously finding subsets $\mathcal{G}_j \subset \mathcal{G}$, eigenspaces $\mathcal{E}_j(\mathcal{G}_j)$ and dimensions p_j. As a result the data in the input set $\mathcal{G}$ is divided into significantly correlated subsets of similar structure, each represented by a separate eigenspace. As we will show in the experiments these eigenspaces correspond to individual activities in accelerometer data and can be used for activity recognition.

The algorithm consists of three phases, namely initialization, eigenspace growing and eigenspace selection. During *initialization*, small subsets of data vectors are chosen from

the input set $\mathcal{G}$, and their respective eigenspaces are calculated and initialized with dimension zero. During *eigenspace growing*, the initial sets are successively enlarged by adding data vectors and accepting or rejecting them based on reconstruction error. At the same time, the corresponding eigenspaces are recomputed and their dimension is adapted. Since the growing process produces overlapping and thus redundant sets and eigenspaces, the final *eigenspace selection* phase applies an optimization procedure that finds a subset of eigenspaces that best represent the data with minimal redundancy. Importantly, the number of eigenspaces that are finally selected is determined automatically during eigenspace selection and does not have to be specified in advance. In the following we describe the three phases of the algorithm in more detail.

4.2.3 Initialization

The input to the algorithm is a set $\mathcal{G} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m | \mathbf{x}_i \in \mathbb{R}^n\}$ containing data vectors which we will refer to as *segments* in the following. During initialization, a large number of small and redundant subsets $\mathcal{G}_j^0 \subset \mathcal{G}$ is generated, uniformly distributed across $\mathcal{G}$. In the extreme case, each segment in $\mathcal{G}$ can serve as an initial subset $\mathcal{G}_j^0$ (as in our examples). For each $\mathcal{G}_j^0$, the corresponding eigenspace $\mathcal{E}_j^0(\mathcal{G}_j^0)$ is calculated, and its dimension p_j^0 is set to zero, i.e., the eigenspace equals the mean of the segments contained in $\mathcal{G}_j^0$.

4.2.4 Eigenspace Growing

After the initial sets $\mathcal{G}_j^0$ have been constructed, they are iteratively enlarged and their corresponding eigenspaces are updated. In the following, $\mathcal{G}_j^t$ and $\mathcal{E}_j^t$ denote the set $\mathcal{G}_j$ and its eigenspace $\mathcal{E}_j$, respectively, at step t of the iteration. p_j^t denotes the dimension of $\mathcal{E}_j^t$ at step t.

The growing process is driven by two error measures, δ_i and ρ_j. δ_i is related to single segments and denotes the reconstruction error $||\mathbf{x}_i - \hat{\mathbf{x}}_i||$ of segment $\mathbf{x}_i$ when projected onto an eigenspace. The second error measure, ρ_j, is related to eigenspaces and defined as the sum of the reconstruction errors of all segments contained in $\mathcal{G}_j$ after projection to $\mathcal{E}_j$. Both δ_i and ρ_j are associated with thresholds that cause the growing process to terminate once the errors get too large.

The term *growing* refers to increasing the size of a set of segments, as well as to increasing the so-called *effective dimension p* of the corresponding eigenspace. The effective dimension denotes the dimension which is necessary to represent the segments in the eigenspace to an adequate degree of accuracy, and it can usually be chosen to be much smaller than the full dimension of the eigenspace.

In step t of the iteration, the following procedure is applied to each set $\mathcal{G}_j^t$: Each segment not contained in $\mathcal{G}_j^t$ is projected onto $\mathcal{E}_j^t$. If a segment's reconstruction error δ_i

is below a threshold, the segment is temporarily accepted into the set $\mathcal{G}_j^{t+1}$. If none of the segments are accepted, the growing for this set is terminated. Otherwise, the new eigenspace $\mathcal{E}_j^{t+1}$ and its aggregate reconstruction error ρ_j are calculated. If the error is below a threshold, the new eigenspace is accepted. Else, the effective dimension p_j of $\mathcal{E}_j^{t+1}$ is increased by one, and the error is recomputed. If the increase in dimension has lowered the error below the threshold, the new eigenspace is accepted. Otherwise, both $\mathcal{G}_j^{t+1}$ and $\mathcal{E}_j^{t+1}$ are reverted to their previous state, and growing of this eigenspace is terminated.

In the worst case, i.e. when all sets $\mathcal{G}_j$ grow until they include all segments in $\mathcal{G}$, the complexity of eigenspace growing is $O(nm^3)$, where n is the dimension of the segments and m the number of segments in $\mathcal{G}$. However, this only holds if there is no or only very little structure in the data. Usually the sets only grow to a fraction of the total number of segments, which significantly reduces computation.

4.2.5 Eigenspace Selection

The result of eigenspace growing is a set of eigenspaces each representing a subset of the input data. The set is redundant in the sense that the subsets overlap in many cases. With respect to robustness of the final outcome this redundancy is an important property of the algorithm.

In the selection step, a subset of the eigenspaces is selected that best represents the data with minimal overlap between the eigenspaces. This is achieved by solving an optimization problem based on the principle of minimum description length (MDL). The goal can be formulated as minimizing the overall description length $L(\mathcal{G})$ of the input $\mathcal{G}$ in terms of eigenspaces:

$$L(\mathcal{G}) \quad = \quad L(\mathcal{M}) + L(\mathcal{G}|\mathcal{M}). \tag{4.3}$$

Here, $L(\mathcal{M})$ denotes the encoding cost of the model, which in our case is the encoding length of all eigenspaces, plus the encoding length of the coefficients y_{kj} for all segments $\mathbf{x}_i \in \mathcal{G}$. $L(\mathcal{G}|\mathcal{M})$ are the costs of specifying the data given the model, which in our case equal the reconstruction errors resulting from the reduced dimension of the eigenspaces.

As noted by [Leonardis *et al.* 2002], minimizing the description length is equivalent to maximizing the savings $S(\mathcal{E}_j(\mathcal{G}_j))$ one obtains from encoding the segments $\mathbf{x}_i \in \mathcal{G}_j$ in terms of the eigenspace $\mathcal{E}_j$ instead of encoding them individually. These savings can be expressed as

$$S(\mathcal{E}_j(\mathcal{G}_j)) \quad = \quad \underbrace{K_0|\mathcal{G}_j|}_{\textit{individual encoding}} \quad - \underbrace{(K_1 p_j + K_2|\mathcal{G}_j|p_j + K_3|\mathcal{G}_j|\rho_j)}_{\textit{encoding with eigenspace}}. \tag{4.4}$$

In this equation, the constant K_0 is related to the cost of encoding a segment in $\mathcal{G}$ without an eigenspace, K_1 is related to the cost of describing an eigenvector, K_2 is related to the average cost of specifying a coefficient, and K_3 is related to the average cost of specifying the error. Using the savings $S(\mathcal{E}_j(\mathcal{G}_j))$, the optimization problem can be solved by maximizing an objective function of the form

$$F(\mathbf{h}) = \mathbf{h}^T \mathbf{C} \mathbf{h} = \mathbf{h}^T \begin{bmatrix} c_{11} & \cdots & c_{1r} \\ \vdots & & \vdots \\ c_{r1} & \cdots & c_{rr} \end{bmatrix} \mathbf{h}, \tag{4.5}$$

where the binary vector $\mathbf{h} = [h_1, h_2, \ldots, h_r]^T$ represents a possible set of eigenspaces, h_j being 1 if the eigenspace j is included in the set, and 0 if not. The diagonal entries $c_{jj}, j = 1, \ldots, r$ of the matrix $\mathbf{C}$ are the savings obtained by the j-th eigenspace, i.e. $c_{jj} = S(\mathcal{E}_j(\mathcal{G}_j))$. The off-diagonal entries c_{jk} penalize overlaps of pairs of sets $\mathcal{G}_j$ and $\mathcal{G}_k$:

$$c_{jk} = |\mathcal{G}_j \cap \mathcal{G}_k|(-K_0 + K_3 \rho_{jk})/2, \tag{4.6}$$

where $|\mathcal{G}_j \cap \mathcal{G}_k|$ denotes the number of segments shared by $\mathcal{G}_j$ and $\mathcal{G}_k$, and ρ_{jk} is the maximum error of the segments in $\mathcal{G}_j \cap \mathcal{G}_k$. Using a greedy algorithm, the optimization problem can be solved in $O(r^2)$ time, where r is the total number of eigenspaces in consideration.

In Section 4.3 we give a detailed example of the different phases of the multiple eigenspace algorithm. Before that, we next propose an extension that allows to analyze data on different time scales.

4.2.6 Extension to Multiple Time Scales

The multiple eigenspace algorithm operates on a single time scale, i.e., all input segments are of the same length. While this property may be acceptable in some domains, for activities it is not obvious which scale or length a data segment should have. Furthermore, as we have seen in Chapter 3, we must assume that there is no single 'best' segment size, as activities happen on different time scales. For these reasons, we extended the algorithm to include multiple scales and allow for different segment sizes.

The extended version of the algorithm accepts as input a signal and a list of n segment sizes. Initialization (Section 4.2.3) and eigenspace growing (Section 4.2.4) are then performed n times. Each time, the signal is divided into signals of a different size. This results in n sets of eigenspaces representing parts of the input at different scales. All of them compete to be included in the final description during a modifed version of the eigenspace selection step. We modified the eigenspace selection so that segments and reconstruction errors on different scales can be compared to each other. In the following we describe the modified selection step in more detail.

Modified Eigenspace Selection

In the selection step of the original approach, cost and savings were defined in terms of entire segments. Since in the modified algorithm, segments can be of different size, we need to redefine the savings in order to make them comparable across different segment sizes. We achieve this by defining the savings in terms of individual samples instead of segments. For a set $\mathcal{G}_j$ containing segments made up of l_j samples each, the savings $S(\mathcal{E}_j(\mathcal{G}_j))$ achieved by encoding the segments in terms of the eigenspace $\mathcal{E}_j$ can be expressed as

$$\begin{aligned} S(\mathcal{E}_j(\mathcal{G}_j)) &= K_0|\mathcal{G}_j| - (K_1 p_j + K_2|\mathcal{G}_j|p_j + K_3|\mathcal{G}_j|\rho_j) && (4.7)\\ &= l_j|\mathcal{G}_j| - l_j p_j - K_2|\mathcal{G}_j|p_j - K_3|\mathcal{G}_j|\rho_j && (4.8) \end{aligned}$$

We thus replaced the constants K_0 (cost of describing a segment without an eigenspace) and K_1 (cost of encoding an eigenvector) by the variable segment length l_j. In the matrix $\mathbf{C}$ of the optimization function (see eq. 4.5) the diagonal terms now represent the adapted savings, $c_{jj} = S(\mathcal{E}_j(\mathcal{G}_j))$, and the off-diagonal entries c_{jk} are redefined as

$$c_{jk} = |\mathcal{G}_j \cap \mathcal{G}_k|(-1 + K_3\rho_{jk})/2 \quad (4.9)$$

where $|\mathcal{G}_j \cap \mathcal{G}_k|$ describes the number of *samples* (before: segments) contained in the intersection of the sets $\mathcal{G}_j$ and $\mathcal{G}_k$.

4.3 Example

Figures 4.2 and 4.3 illustrate the different phases of the multiple eigenspace algorithm for a single time scale. The upper part of Figure 4.2 shows an acceleration signal, recorded by an accelerometer attached to the wrist of a user while juggling 3, 4 and 5 balls, respectively. The signal was divided into segments of four seconds and transformed to the frequency domain before applying the algorithm.

The bottom plot of Figure 4.2 shows the result of the growing phase, i.e. the sets $\mathcal{G}_j$. The horizontal axis represents the segments into which the signal was divided, and the vertical axis corresponds to the sets $\mathcal{G}_j$. In row j, the segments belonging to the set $\mathcal{G}_j$ are marked in gray (e.g., $\mathcal{G}_8$ consists of segments 6 to 11). Three sets were chosen during the final selection procedure, they are highlighted in the figure. Note that there are only a few sets of segments across the borders of the three juggling patterns, and the final sets match the three patterns closely.

Figure 4.3 illustrates the eigenspace growing process for the example in Figure 4.2: Initially, each set is made up of one segment. As the growing proceeds, one can observe three groups of sets forming along the three parts of the signal. Finally, during eigenspace selection (see sec. 4.2.5), one set of each of those groups gets selected. Figure 4.4 shows the effect of using different error thresholds δ and ρ.

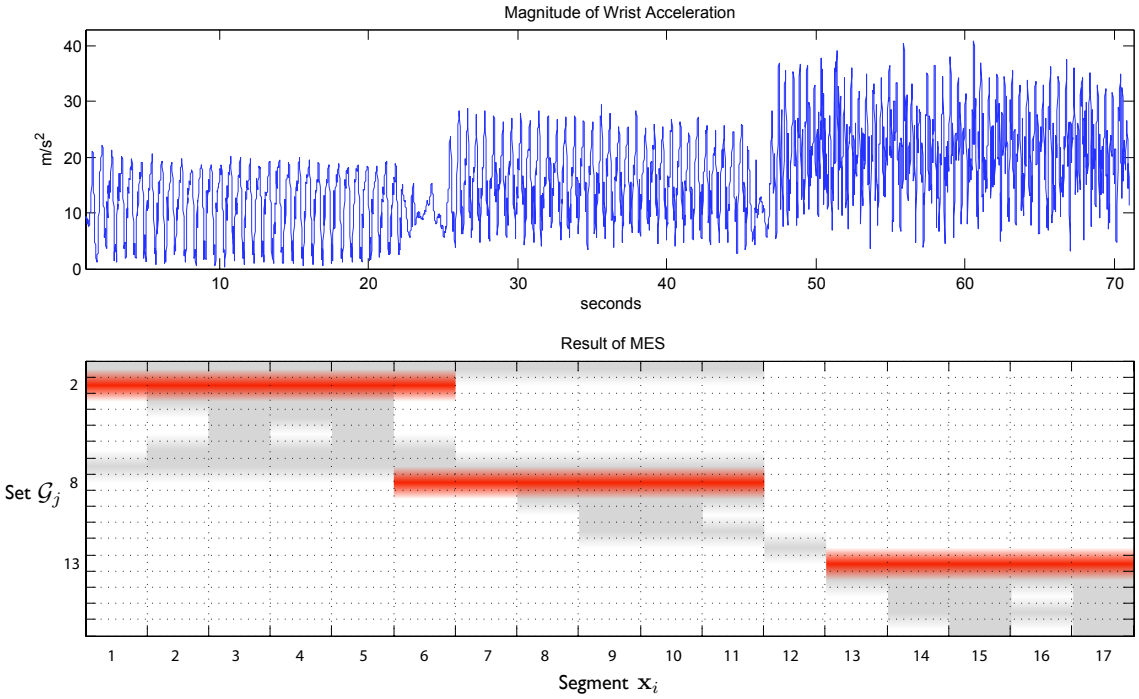


***Figure 4.2**: Application of the multiple eigenspace algorithm to an acceleration signal. Top: Magnitude of wrist acceleration. Bottom: The result of eigenspace growing. The sets $\mathcal{G}_i$ are marked, and those that were finally selected ($\mathcal{G}_2, \mathcal{G}_8$ and $\mathcal{G}_{13}$) are highlighted.*

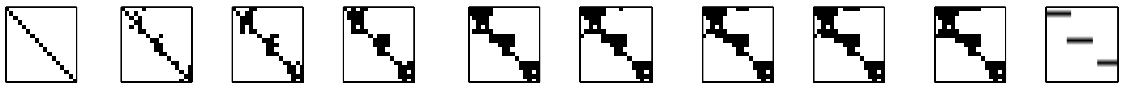
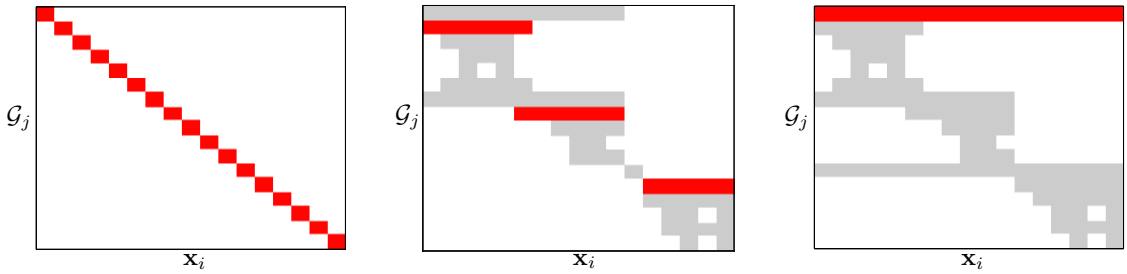


***Figure 4.3**: Eigenspace growing and selection corresponding to the data in Figure 4.2. From left to right, different stages of the growing process are shown. The rightmost plot shows the result of the selection step.*

4.4 Initial Experiments

After describing the multiple eigenspace algorithm and giving an example in the previous section, we now demonstrate the feasibility of our approach for analyzing sequences of activity data. First, we briefly introduce the sensor platform and the data sets used for our experiments. Then, we discuss two possible methods of applying the algorithm: using raw acceleration data on multiple time scales and using FFT features on a single time scale. Finally we compare the two feature representations in terms of their classification performance. While for the results of this section a relatively short recording of different walking modes was used, the next section will report on results obtained from longer recordings of mixed activity data.



(a) (b) (c)

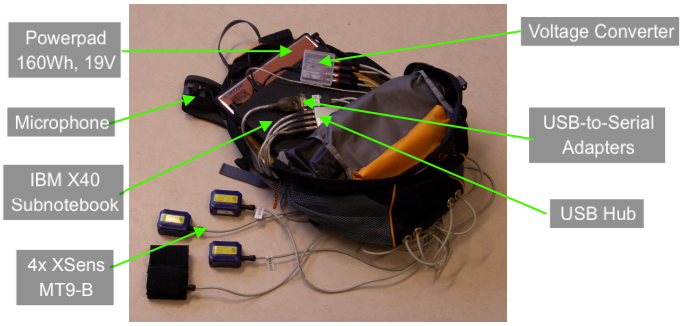
Figure 4.4: *Effect of varying the error thresholds δ and ρ that control acceptance of new segments $\mathbf{x}_i$ into sets $\mathcal{G}_j$ during the eigenspace growing phase (based on the data shown in fig. 4.2). Low thresholds lead to small sets (a), and high thresholds may lead to large sets that cover all of the data (c). In (b) we show the results for $\delta = \rho = 3.5 * 10^{-3}$, which we found to work well for all datasets that we used. Set membership is indicated in gray, and the sets determined by eigenspace selection are highlighted.*

4.4.1 Sensor Platform

Figure 4.5(a) shows the sensor platform used for the experiments. The main components are four inertial sensors connected to an IBM X40 laptop via a USB hub. The laptop, together with batteries and adapters, is situated in a small backpack carried by the user, and the inertial sensors are worn by the user, e.g. on wrist, hip, ankle or other parts of the body. The recording software runs on the laptop and can be remote-controlled from a PDA. As inertial sensors we initially used the model MT9-B by Xsens and later the model MTx, which features a larger measurement range and better on-board processing capabilities. Besides 3-D acceleration, the sensors output 3-D rate of turn and 3-D magnetic field data, as well as an absolute orientation estimate. For annotation purposes, we also record audio data using a stereo microphone clipped to the shoulder strap of the backpack. Figure 4.5(b) shows the entire sensor platform worn by a user.

4.4.2 Data Set

The sensor platform described above was used to record data of various activities, ranging in length from several seconds to about thirty minutes. During recordings, the inertial sensors were attached to wrist, hip, thigh and ankle of the user. In the experiments we report, we focus on the data from wrist and hip, as preliminary experiments showed that these two locations were discriminant enough for our set of activities. For the initial experiments, walking modes of different speeds were recorded separately. Subsequent recordings consist of a mix of several activities, including different walking modes, climbing stairs and juggling different numbers of balls.



(a) The sensor platform used for the experiments.

(b) User with sensors on hip, thigh, wrist and ankle.

***Figure 4.5**: Sensor Platform*

4.4.3 Experiments with Multiple Time Scales

Figure 4.7 shows the result of applying the extended multiple eigenspace algorithm to the signal shown in Figure 4.6, using three different segment sizes. The signal consists of the acceleration magnitude measured at the hip, covering three different modes of walking (*walking at normal pace*, *jogging* and *walking fast*) and sampled at 200 Hz for about one minute. The sets corresponding to the four eigenspaces chosen by the selection step of the algorithm are highlighted. The topmost covers the entire signal, while the remaining three each represent segments that correspond to the three walking modes, respectively. Each of those three eigenspaces is based on a different segment length.

We found that in order to obtain eigenspaces that represent activities well, the underlying segment lengths need to match the periodicity of the data closely. Thus, in order to obtain satisfying results, one has to carefully choose segment lengths, e.g. based on the periodicity of the signal. This makes the approach rather inflexible. Furthermore, since we are interested in finding structure in an unsupervised fashion, we cannot assume that we know about the periodicity or other properties of the data in advance. To address these issues, we changed our features from raw signal data to frequency components, which we will discuss in the next section. Apart from that, we believe that the proposed extension of the multiple eigenspace algorithm to multiple time scales is a general scheme that can be applied to any kind of data, and which allows simultaneous analysis of data at different time scales.

4.4.4 Experiments in Frequency Space

We conducted a series of experiments using FFT coefficients computed over the acceleration signal as features, with the goal of obtaining a representation of the data that does

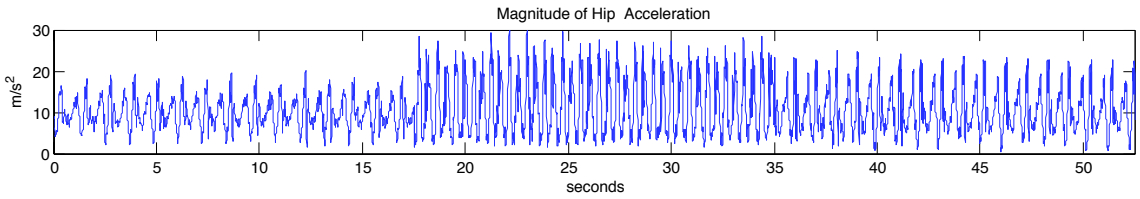


***Figure 4.6**: Constructed dataset, consisting of three different walking modes (left third: walking, middle third: jogging, right third: walking fast). Shown is the magnitude of the acceleration measured at the hip, sampled at 200 Hz.*

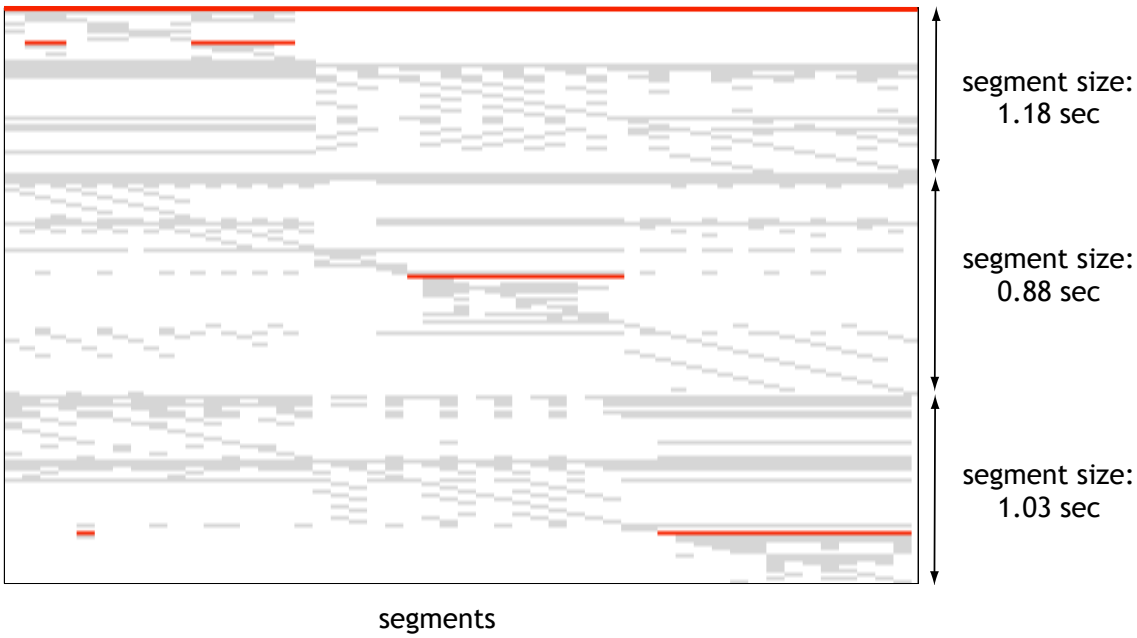
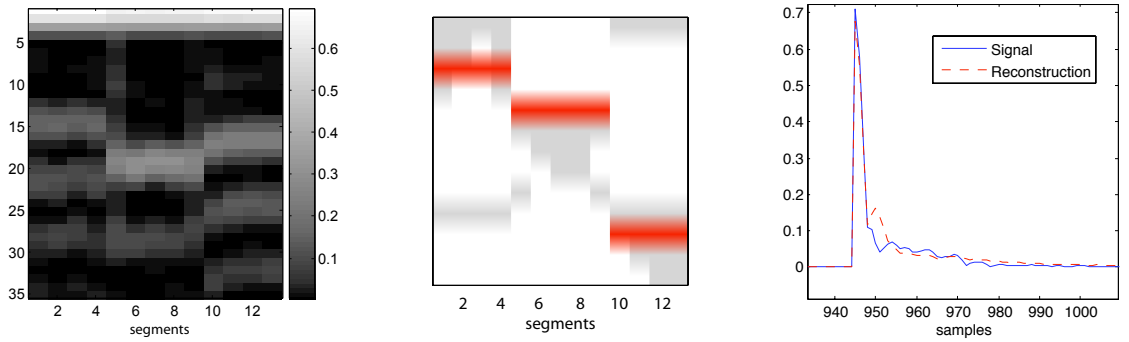


***Figure 4.7**: Result of applying the adapted algorithm to the signal shown in Figure 4.6. Three different segment sizes between 0.88 and 1.18 seconds were used, and four eigenspaces were selected, which are highlighted in the figure.*

not require a priori knowledge about properties such as the periodicity of the signal. For these experiments we applied the multiple eigenspace algorithm on single time scales. We found that FFT features computed over a single scale can be used effectively to separate different activities using multiple eigenspaces. However, the choice of the segment length involves a tradeoff between short segments that capture basic activities but might yield unstable FFT results, and longer segments which yield more stable results but might be too long to allow discrimination between basic activities.

During the experiments, we found that segment sizes of around 4 seconds lead to good results when using FFT features in combination with our set of activities. Figure 4.8 shows the result of applying the multiple eigenspace algorithm to the FFT coefficients computed over the signal shown in Figure 4.6. Figure 4.8(a) shows the spectrogram, the vertical axis corresponding to the first 35 FFT coefficients, the horizontal axis to the



(a) Spectrogram

(b) Sets $\mathcal{G}_j$

(c) Signal and reconstruction of a single segment

***Figure 4.8**: Application of the multiple eigenspace algorithm to data of three different walking modes (see fig. 4.6). As features, FFT coefficients computed over windows of four seconds were used.*

segments of the signal. For all three walking patterns, most of the energy is contained in the first three coefficients, however each activity has a distinct and consistent distribution of peaks in the rest of the spectrum. This structure is captured well by the eigenspaces, as can be seen from Figure 4.8(b). Three eigenspaces are selected, each corresponding to a different walking pattern. Figure 4.8(c) shows one of the feature segments for the activity *walking at normal pace* and its reconstruction, which differs only slightly from the original.

These initial experiments led us to believe that using multiple eigenspaces on features in frequency space is a promising approach to detecting structure in more diverse sets of activities. Before discussing such experiments in Section 4.5, we will first report on some initial classification results in the next section.

4.4.5 Classification

The eigenspaces obtained from the algorithm can be used as classifiers for activities, based on the reconstruction error of unknown data segments. To classify a segment, it is projected onto each eigenspace and then assigned to the one that yields the lowest reconstruction error. When using a sliding window, we classify individual samples using segments that end at the sample. In the following we compare the classification performance of models based on signal- and FFT-features.

In Figure 4.9, two runs of the multiple eigenspace algorithm on the walking patterns, with subsequent classification, are compared side by side. Figure 4.9(a) shows the result when using the adapted version with multiple time scales on the plain acceleration signal. The bottom plot shows the reconstruction error of the signal for all five models (eigenspaces) that were selected. For each model, the reconstruction error was computed over a sliding window of the same size as the segment size of the model, and shifted over

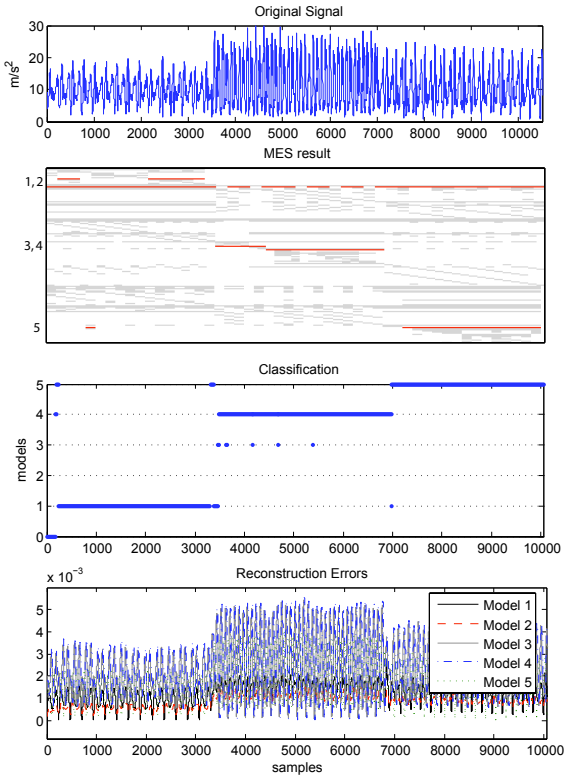
the signal in steps of single samples. As a consequence, the error is smallest when the window is aligned with the segment positions at construction time of the models, and largest when shifted by 50%. This results in an oscillating reconstruction error with a period of the segment size of the eigenspace. Figure 4.9(c) shows a close-up view of the reconstruction errors in Figure 4.9(a). To avoid that the oscillating errors are reflected in an unstable classification result, we performed a smoothing by classifying each sample by the model with the lowest error over a window of preceding samples. This leads to the classification results shown in Figure 4.9(a). *Walking normal*, *jogging* and *walking fast* are assigned to model (i.e. eigenspace) 1, 4 and 5 respectively. Furthermore, one can observe that model 2, which represents large parts of the signal as a result of the algorithm, is outperformed in terms of reconstruction error by other, more specialized models throughout the length of the signal.

Even though the classification results using plain acceleration data are acceptable, the sensitvity of the reconstruction error to the position of the sliding window advises against using the raw signal as feature. In contrast, Figure 4.9(b) shows that in the frequency domain, similar (sample-based) classification results can be obtained without the need to smooth out the error curve. The bottom plot of Figure 4.9(b) shows that for models based on FFT features, the curves of the reconstruction errors are smooth and stable within each of the three parts of the signal. This implies that this approach is insensitive to shifts between the sliding window and the segment boundaries at the time of model construction. Moreover, for each part of the signal, the errors are in distinct order. Altogether, these properties result in a more robust classification. As a consequence, we only consider FFT features in the remaining experiments.

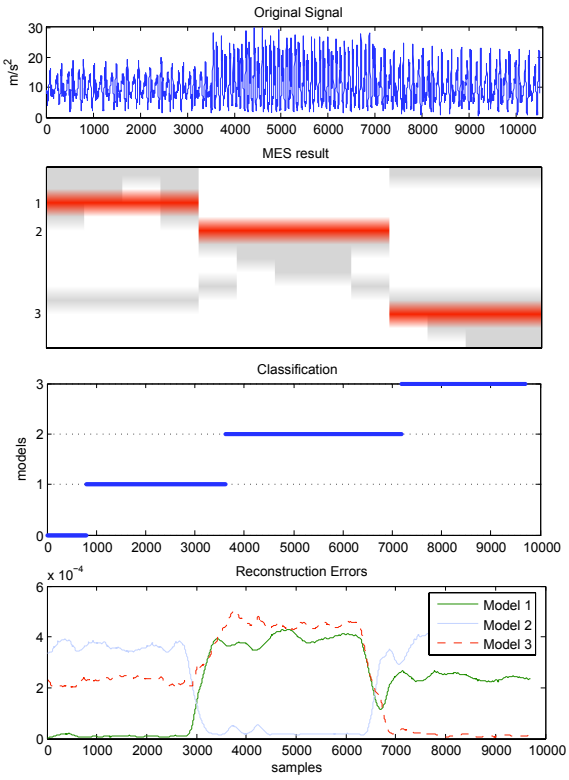
4.5 Experiments with Mixed Activity Data

In this section, we show how our approach finds structure in real-world recordings that cover a number of different activities. The data was recorded using the sensor platform described in Section 4.4.1, and two inertial sensors were worn on wrist and hip of the user. We will first report on the results for hip and wrist individually, and then show how performance can be improved by combining the models of both recordings. The recordings lasted for about a quarter of an hour. As features we use FFT coefficients, computed over windows of 4 seconds, as this combination had proven to yield the best results in the above experiments. The feature vectors were normalized to length 1 before being passed to the multiple eigenspace algorithm.

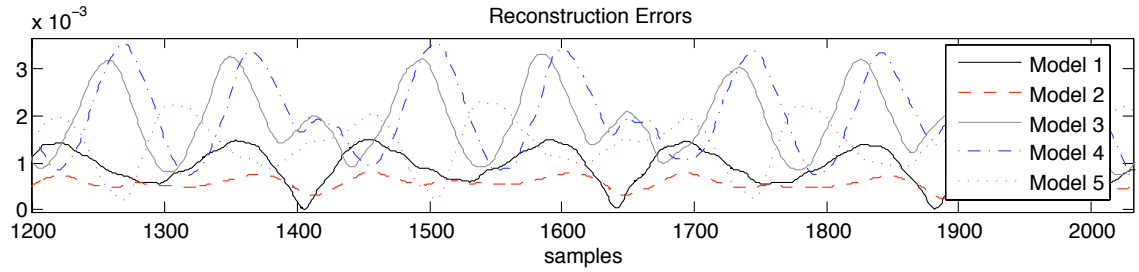
The recording shown in Figure 4.10 contains six different activities: *walking*, *walking upstairs*, *walking downstairs*, and juggling 3, 4 and 5 balls, respectively. The top of the figure shows the raw signal, which in this case is the magnitude of the acceleration occurring at the hip, sampled at 100 Hz. The middle plot shows manual annotations, i.e. ground truth for the data. Applying the multiple eigenspace algorithm to this data resulted in seven eigenspaces. With these eigenspaces we performed a classification of the training



(a) Classification based on plain acceleration signal



(b) Classification based on FFT features



(c) Close-up view of the reconstruction errors shown at the bottom of Figure 4.9(a).

Figure 4.9: *Comparison of two different feature representations. Figures 4.9(a) and 4.9(b) show, from top to bottom: acceleration signal; result of applying the multiple eigenspace algorithm (the selected models are numbered); classification based on reconstruction error; reconstruction errors of the different models (i.e. eigenspaces).*

data based on reconstruction error, the same way as described in Section 4.4.5. The result is shown at the bottom of Figure 4.10. Samples that were not assigned to any model because of too large reconstruction errors appear in the row labeled with 0.

When comparing the ground truth to the model assignments in Figure 4.10 , one can observe that the structure is visually similar. On closer inspection, one can see that the activity *walking* is mainly represented by models 4, 5 and 6. *Walking upstairs* corresponds to model 1, and *walking downstairs* to models 2 and 3. The juggling sequences are all assigned to a single model (7), which is not surprising, since there is only very little (and thus nondiscriminative) hip movement during juggling.

In order to judge the quality of the model assignments, we manually chose for each activity the models that best represented them and computed recall and precision values for each set of models representing an activity. The result for the data in Figure 4.10 is shown in the column labeled *Hip* in Table 4.1. The models for *walking* (4, 5 and 6) reach precision and recall values close to 100% (0.99 and 0.98 respectively), followed by *walking downstairs*(1.0/0.93) and *walking upstairs* (0.80/1.0). As there is only one model (7) for the three juggling activities, the table contains only one entry for all three, which stands for the activity *juggling* (0.32/1.0).

Figure 4.11 shows a second set of acceleration data, recorded at the wrist. The ground truth is the same as for Figure 4.10. Fewer models were selected this time, but they describe the data more precisely than the models for the hip recording – there is a significant gain in the average precision over time (from 0.57 to 0.76) and only a slight reduction in recall (from 0.98 to 0.92). The increase in precision is due to the fact that the juggling patterns can be discriminated at the wrist.

In Figure 4.12, a combination of the models from the wrist and hip recordings is used for classification. Each model from the wrist recording was combined with each model from the hip recording to form a new model, which makes for 7*8 = 56 models (the *not-assigned* cases were included as model 0). The result can be seen at the bottom of Figure 4.12. Overall precision and recall are now above 90% (0.93/0.97). Compared to the hip recording this means a slight decrease in recall, but on the other hand, the three juggling patterns can now be separated. When comparing to the wrist recording, one can observe significant increases in the precision values for *juggling 4 balls* (0.04 to 0.84) and *walking upstairs* (0.49 to 0.84).

4.6 Conclusion

An important argument made in this chapter is that unsupervised techniques for activity recognition are highly desirable. To this end we haved proposed a novel approach to discover structure in sensor data of human activity in an unsupervised fashion. We demonstrated the feasibility of the approach by applying it to acceleration data recorded from body-worn sensors. For the set of activities analyzed, our system was able to build

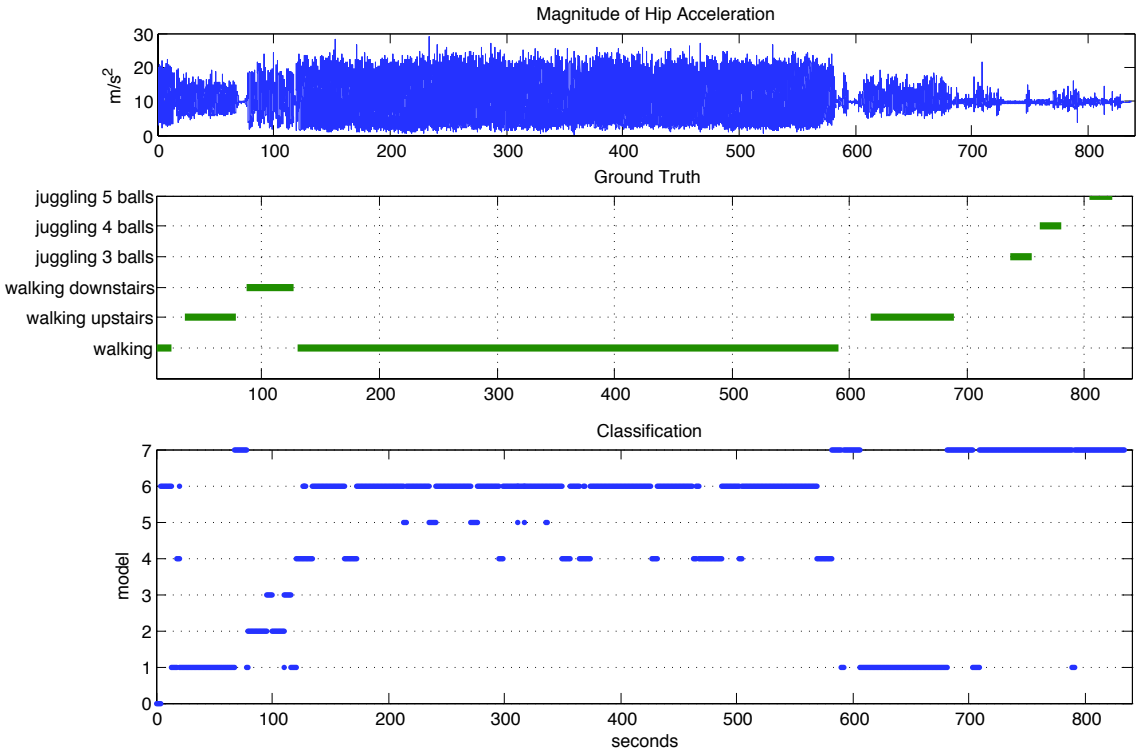


***Figure 4.10**: Recording of approx. 14 min length, magnitude of **hip** acceleration. From top to bottom: raw signal, ground truth, and classification based on seven models constructed by the multiple eigenspace algorithm.*

Activity	Data Set		
	Hip	Wrist	Combined
Walking	0.99/0.98	0.93/0.97	0.99/0.97
Walking upstairs	0.80/1.00	0.49/1.00	0.84/1.00
Walking downstairs	1.00/0.93	0.95/0.45	0.98/0.93
Juggling 3 balls	0.32/1.00	0.76/1.00	0.82/1.00
Juggling 4 balls		0.04/1.00	0.84/1.00
Juggling 5 balls		0.60/1.00	0.60/1.00
Average over time	0.57/0.98	0.76/0.92	0.93/0.97

***Table 4.1**: Precision/Recall for different activities and data sets*

models that correspond to different activities, without requiring any prior training, user annotation or information about the number of activities involved. When used for classification, the system shows recognition rates comparable to other, supervised approaches. We found that for acceleration data of basic activities such as walking, using frequency components as features results in models that can represent the different activities well and that can be used for robust classification. Finally, we showed that classification rates

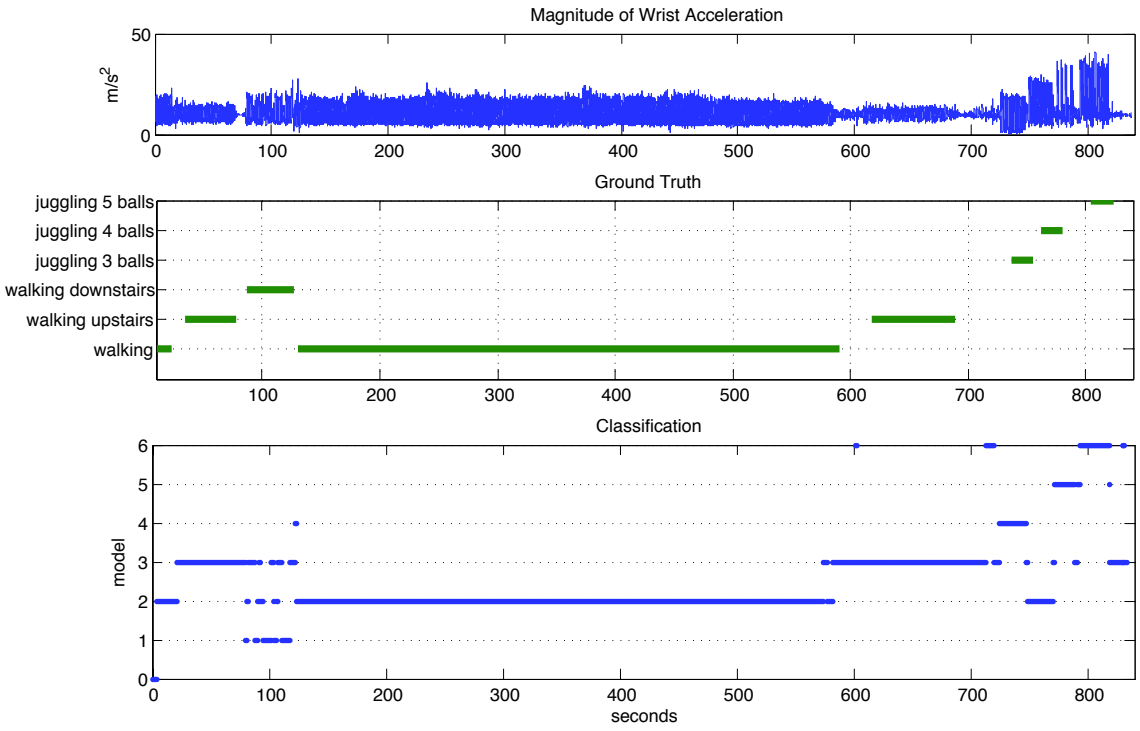


***Figure 4.11**: Recording of approx. 14 min length, magnitude of **wrist** acceleration. From top to bottom: raw signal, ground truth, and classification based on six models constructed by the multiple eigenspace algorithm.*

can be improved when combining the data of two sensors located at different body locations.

An aspect that is appealing for activity recognition is that the number of activities does not need to be specified beforehand. Instead, the convergence of the method is controlled mainly via the sample inclusion thresholds. In practice, we found that it suffices to use a single parameter that is coupled to the thresholds via constant factors.

Obviously, the results presented in this chapter are only a first step towards unsupervised discovery of activities in arbitrary sensor data. As pointed out before, however, the multiple eigenspaces approach is general in the sense that in principle it can handle different sensor modalities and different types of activities. In the next chapter, we will focus on the generative nature and the classification capabilities of the approach, and show that in combination with discriminant learning, high recognition rates can be obtained with only little supervision.

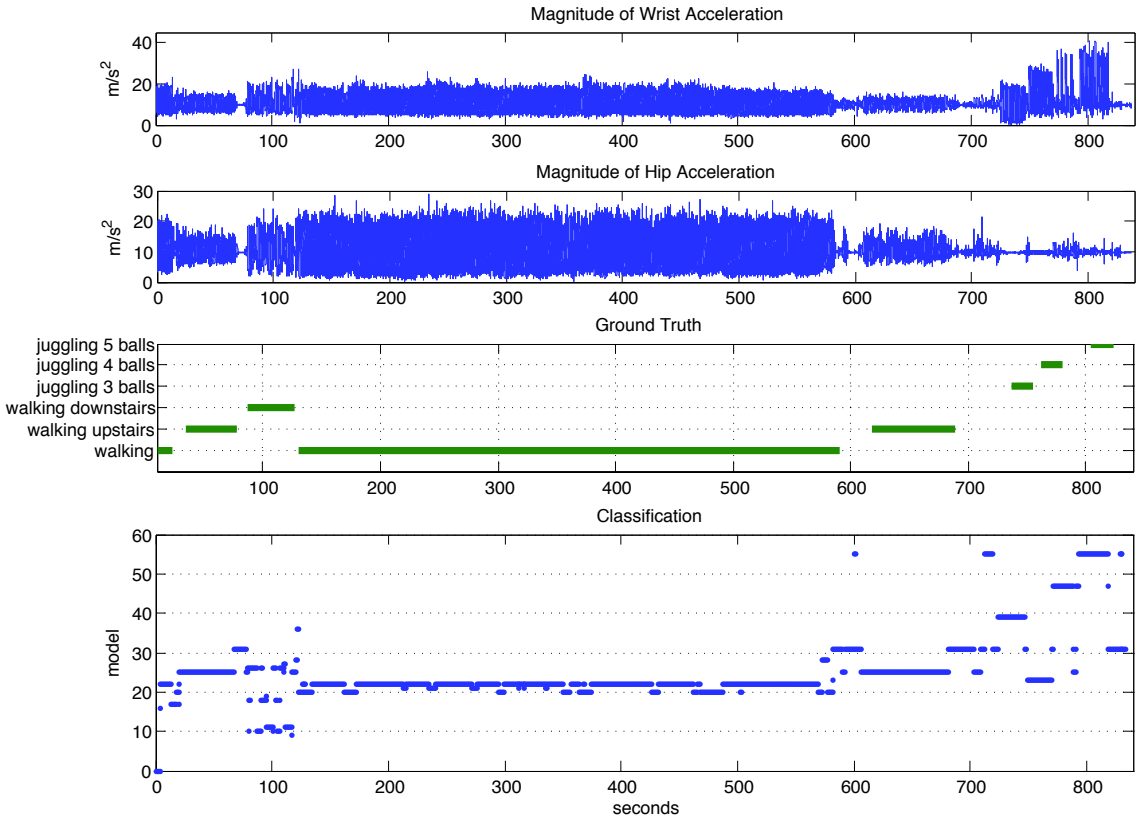


Figure 4.12: *Classification based on the combination of the models for hip (fig. 4.10) and wrist (fig. 4.11).*

5

Combining Discriminative and Generative Learning

State-of-the-art activity recognition algorithms can roughly be divided in two groups concerning the choice of the classifier, one group using generative models and the other discriminative approaches (see Section 2.5.2 for a discussion). This chapter presents a method for activity recognition which combines a generative model with a discriminative classifier in an integrated approach. The generative part of the algorithm allows to extract and learn structure in activity data without any prior labeling or supervision. The discriminant part then uses a small but labeled subset of the training data to train a discriminant classifier. In experiments we show that this scheme enables to attain high recognition rates even though only a subset of the training data is used for training. Also the tradeoff between labeling effort and recognition performance is analyzed and discussed.

5.1 Introduction

In this chapter we integrate two different types of approaches into a single common framework to fully exploit their strengths while minimizing their weaknesses. More specifically, we combine a generative model (multiple eigenspaces) with SVM training on partially labeled training data. The idea of using a generative model inside a kernel function has been proposed before [Jaakkola and Haussler 1998, Jebara *et al.* 2004, Vasconcelos *et al.* 2004, Tsuda *et al.* 2002] and a similar idea has been applied to activity recognition [Lester *et al.* 2005]. However, these approaches do neither address nor analyze the issue of reducing the amount of supervision and labeled training data.

The first main contribution of this chapter is the combination of the generative model of multiple eigenspaces with a discriminant SVM classifier into a single activity recognition framework. On the one hand the new integrated approach allows to significantly increase recognition accuracy w.r.t. to the multiple eigenspace approach by rejecting false positives more effectively. On the other hand the approach allows to train discriminant classifiers on only part of the data and therefore to substantially reduce the amount of supervision required. The second main contribution are experimental results which show the

superiority of the new integrated approach, both with respect to the multiple eigenspace approach and with respect to a baseline system using a Naïve Bayes classifier. The performance is analyzed in particular with respect to the amount of labeled training data, and the tradeoff between supervision and recognition accuracy is shown experimentally.

The rest of the chapter is organized as follows. Section 5.2 gives a short introduction of the multiple eigenspace approach (for details, see Chapter 4). Section 5.3 then introduces the integrated approach that uses multiple eigenspaces and discriminant training in order to learn discriminant classifiers on partially labeled data. Section 5.4 consists of three sets of experiments to analyze the performance of the integrated approach as well as to analyze its performance when the amount of supervision is reduced. Section 5.5 concludes the chapter.

5.2 Multiple Eigenspaces

As we have seen in Chapter 4, the multiple eigenspace algorithm is a general procedure to extract and represent low-dimensional structure from high-dimensional input data [Leonardis *et al.* 2002]. It is based on principal component analysis (PCA), a common technique in pattern recognition to reduce the dimensionality of feature spaces (see [Duda *et al.* 2004], for example). While PCA finds a single eigenspace that best represents all input features in a least-squares sense, the multiple eigenspace approach finds several of such eigenspaces, each representing a highly correlated subset of the input data. The advantage of this approach is that the dimensionality of the resulting eigenspaces can be much lower than when using a single eigenspace. More importantly, the eigenspaces can serve models for correlated subsets of the data. As we have seen in Chapter 4, such models can be used to detect and represent structure such as individual activities in accelerometer data. In this chapter we focus on the generative nature of the algorithm and its ability to discover structure in data without supervision.

Example. Figure 5.1 shows an example of applying the multiple eigenspace algorithm to features computed from twelve body-worn accelerometers (the features and dataset are described in detail in section 5.4.1). The upper plot of Figure 5.1 shows the features, and the middle plot shows the ground truth of the recording, consisting of eight different activities the user was performing while wearing the sensors.

Sixteen eigenspaces were chosen in the final selection phase. Each eigenspace is a representative model for a subset of the input, and these models often correspond to activities of the user. This can be seen from the bottom plot of Figure 5.1, in which we assign to each segment the eigenspace which has the lowest reconstruction error with respect to the segment. E.g., eigenspace 16 largely corresponds to activity 6 (*writing on the whiteboard*), eigenspace 14 to activity 1 (*standing*), and eigenspace 2 to activity 2 (*walking*).

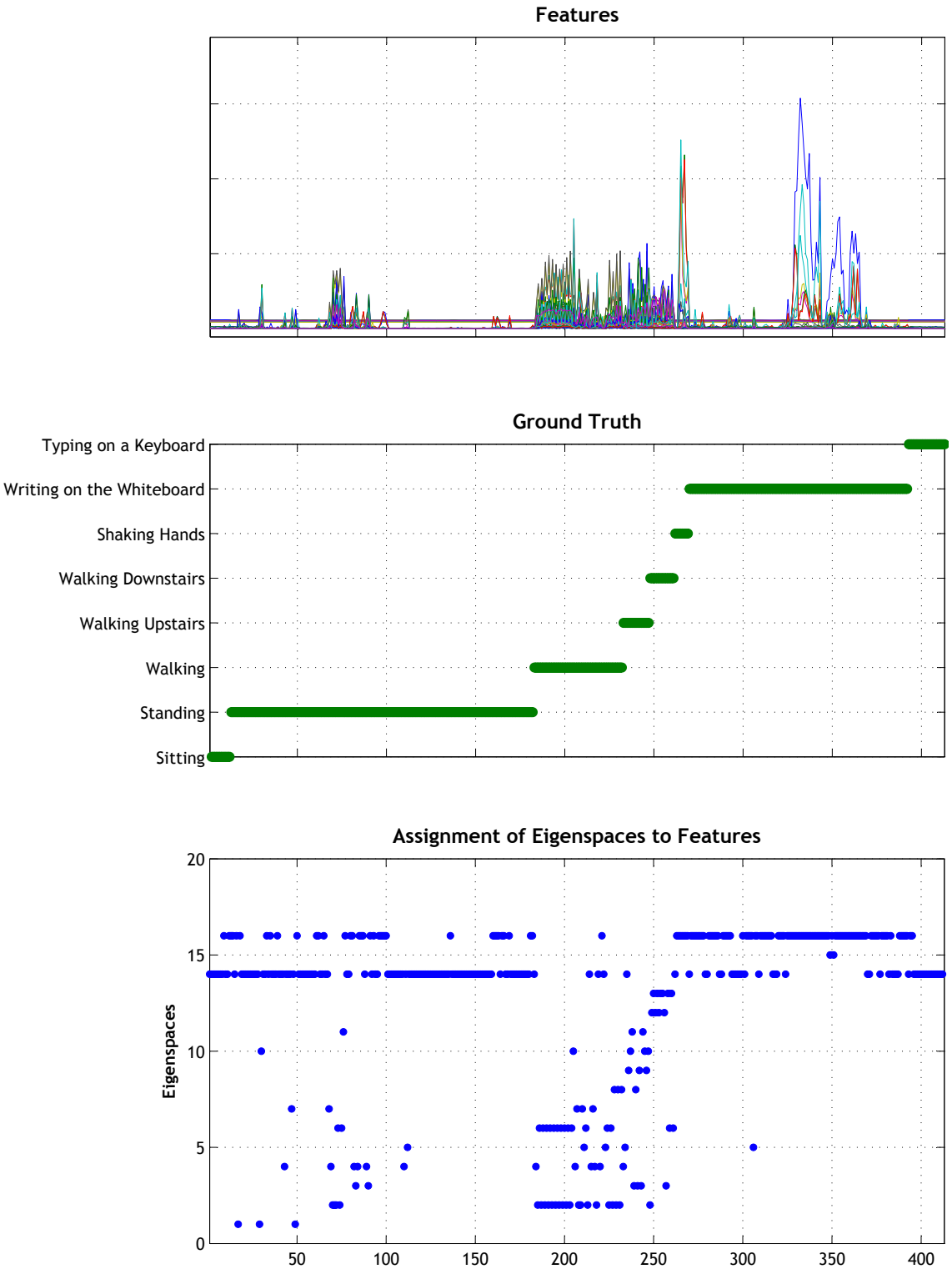


***Figure 5.1**: Application of the multiple eigenspace algorithm to features computed from twelve body-worn accelerometers. Top: mean and variance of the acceleration signals. Center: ground truth of performed activities. Bottom: assignment of eigenspaces to samples, based on reconstruction error.*

Even though there is seldom a perfect correspondence between models and activities, one can clearly see that there is a high correlation between several models and activities. This fact can be used to turn the models into classifiers, by associating each model with the activity that occurs most often in the subset covered by the model. We report on the results of the classification in Section 5.4. Before that, we describe in Section 5.3 how we combine the multiple eigenspace algorithm with a discriminative classifier, namely support vector machines.

5.3 Combining Multiple Eigenspaces with Support Vector Machines

In Chapter 4 we have seen that the concept of multiple eigenspaces can be exploited for unsupervised discovery of structure in activity data [Huỳnh and Schiele 2006b]. Interestingly, the system was able to build models that correspond to different activities without requiring any prior training, user annotation or information about the number of activities involved. When used for classification, the system showed recognition rates comparable to other, supervised techniques.

However, the approach of multiple eigenspaces suffers – as many other generative models – from a significant number of false positives. In order to improve recognition performance there is a clear desire to learn and incorporate discriminant information through a discriminant classification scheme. In this chapter we adopt the support vector framework as it has shown competitive performance on a wide range of different classification tasks.

Obviously, the incorporation of a discriminant classifier requires to use labeled training samples. However, an important emphasis of the integrated approach is to keep the amount of required supervision to a minimum. Therefore, during training the approach leverages on the ability of multiple eigenspaces to learn in an unsupervised fashion an intermediate representation and structure description of the sensor data. This allows to use relatively small amounts of supervision while still obtaining competitive recognition performance.

5.3.1 Support Vector Machines

In the following we briefly describe classification with Support Vector Machines (SVMs). Further details can be found e.g. in [Vapnik 1998]. Consider the problem of separating a set of training data $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_1, y_1)$ into two classes, where $\mathbf{x}_i \in \mathbb{R}^N$ is a feature vector and $y_i \in \{-1, +1\}$ its class label. If we assume that the classes can be separated by the hyperplane $\mathbf{w} * \mathbf{x} + b = 0$, and that we have no prior knowledge about the data distribution, then the optimal hyperplane (i.e., the one with the lowest bound on the expected generalization error) is the one with the maximum distance to the closest points

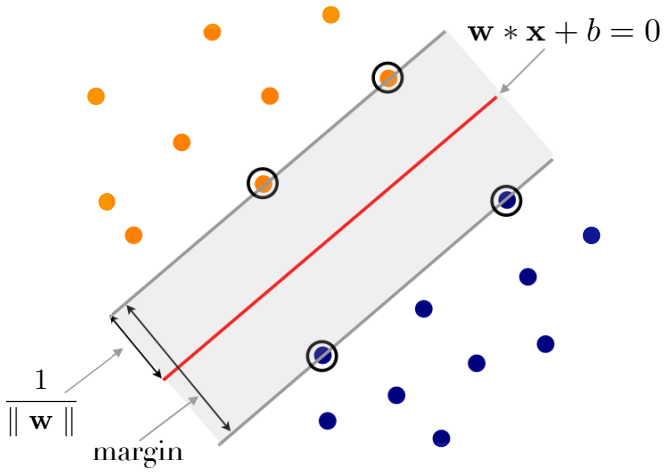


***Figure 5.2**: Example of a support vector classifier in the case where the two classes are linearly separable by a hyperplane $\mathbf{w} * \mathbf{x} + b = 0$. SVMs find parameters $\mathbf{w}$ and b so that the* margin *that separates the two classes is maximized.*

in the training set. The optimal values for $\mathbf{w}$ and b can be found by solving the following constrained minimization problem:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2, \quad subject\ to \quad y_i(\mathbf{w} * \mathbf{x}_i + b) \geq 1, \forall i = 1, \ldots, m \tag{5.1}$$

This is equivalent to maximizing the so-called *margin* that separates the two classes. Figure 5.2 illustrates this concept for a simple example in which the classes are separable by a hyperplane $\mathbf{w} * \mathbf{x} + b = 0$. Solving 5.1 using Lagrange multipliers $\alpha_i (i = 1, \ldots, m)$ results in a classification function

$$f(\mathbf{x}) = sign\left(\sum_{i=1}^{m} \alpha_i y_i \mathbf{w} * \mathbf{x} + b\right). \tag{5.2}$$

where α_i and b are found using an SVM learning algorithm [Vapnik 1998]. Most of the α_i take the value of zero. Those $\mathbf{x}_i$ with nonzero α_i are the so-called *support vectors*. In cases where the classes are non-separable, the solution is identical to the separable case with a modification of the Lagrange multipliers to $0 \leq \alpha_i \leq C, i = 1, \ldots, m$, where C is the penalty for misclassification.

To obtain a nonlinear classifier, one maps the data from the input space $\mathbb{R}^N$ to a high dimensional feature space $\mathcal{H}$ by $\mathbf{x} \rightarrow \Phi(\mathbf{x})$, such that the mapped data points of the two

classes are linearly separable in the features space. Assuming there exists a kernel function K such that $K(\mathbf{x},\mathbf{y}) = \Phi(\mathbf{x}) * \Phi(\mathbf{y})$, a nonlinear SVM can be constructed by replacing the inner product $\mathbf{w} * \mathbf{x}$ by the kernel function $K(x,y)$ in equation 5.2. This corresponds to constructing an optimal separating hyperplane in the feature space. Kernels commonly used include polynomials $K(\mathbf{x},\mathbf{y}) = (\mathbf{x} * \mathbf{y})^d$ and the Gaussian Radial Basis Function (RBF) kernel $K(\mathbf{x},\mathbf{y}) = exp\{-\gamma\|\mathbf{x}-\mathbf{y}\|^2\}$. For our experiments we used the RBF kernel.

The extension of SVMs from 2-class to N-class problems can be achieved e.g. by training N SVMs, each separating a single class from all remaining classes.

5.3.2 Combining Multiple Eigenspaces with SVMs

In order to train an SVM with minimal labeled training data we proceed in two steps. In the first step we use the multiple eigenspace approach to obtain a description of the data that captures the essential structure of the activity data. As the multiple eigenspace approach is fully unsupervised, we can use all sensor data that is available to us, whether labeled or not. The use of unsupervised training data is essential and a desired feature of the approach as this allows to learn from far more data than in the supervised case to derive a good representation and description of the sensor data. Essentially, the approach of multiple eigenspaces provides a low-dimensional description of the sensor-data which lends itself to training SVMs.

In the second step we use the obtained eigenspaces to construct features for training the discriminant SVMs. More specifically, we calculate for each labeled training segment a vector $\mathbf{d} = d_1,\ldots,d_n$ of reconstruction errors. The element d_i is the error of the sample with respect to the eigenspace i, introduced earlier. This corresponds to a soft assignment of samples to eigenspaces, as opposed to the hard assignment in the previous chapter, where each sample was assigned to exactly one eigenspace. By doing this we provide the SVMs with more information for their classification task.

Before feeding the error vectors and the corresponding labels to the SVM, we scale each component d_i of the vectors by

$$\hat{d}_i = e^{-\frac{d_i^2}{\sigma^2}} \tag{5.3}$$

The goal of this transformation is to obtain feature values between 0 and 1 with the property such that the distribution of small (and discriminant) errors is preserved and the errors beyond a certain threshold are mapped to zero. The threshold can be controlled by adjusting the value of σ in equation 5.3, which is related to the variance of the transformation function.

We use a radial basis function as kernel for the SVMs. The parameters C and γ (γ corresponds to the width of the RBF-kernel) were determined by sampling the parameter space on an exponential grid.

During classification, we assign to each test sample a vector **d** of reconstruction errors and transform this vector in the same fashion as done for the training. We then pass the transformed vector to the SVMs for classification.

5.4 Experiments

The following experiments were designed to investigate two main questions. First, how does the performance of the new integrated approach of multiple eigenspaces and SVM-training compare to its predecessor, namely multiple eigenspaces, and to naïve bayes classification as a baseline. And second, what is the effect of decreasing the amount of labeled training data on the three different types of learning algorithms.

After introducing the data set in section 5.4.1, the following sections describe three sets of experiments, each set for a different classification scheme. The first experiment (section 5.4.2) describes and analyses the naïve Bayes approach as a baseline and example of a classical supervised approach. The second set of experiments (section 5.4.3) uses the unsupervised and generative approach of multiple eigenspaces described above. The third set of experiments (section 5.4.4) uses the integrated approach that combines the generative nature of multiple eigenspaces with SVM learning on subsets of training data.

Each experiment was performed in five different configurations. In each configuration we change the amount of labeled training data while leaving the size of the test data unchanged. We start by dividing the entire dataset into 80% training and 20% test set. Then we gradually decrease the amount of labeled training data, from 80% down to 5% of the entire dataset, while leaving the size of the test set fixed. For the naïve Bayes classifier, this means that the size of the available training data is reduced in each iteration, as it cannot learn from unlabeled data. The multiple eigenspace approach however can still be trained on the unlabeled data. It uses the reduced set of labeled training data only for learning a mapping from labels (i.e. activities) to the models it has constructed from the unlabeled data. Similarly the integrated approach uses the unlabeled data to train the multiple eigenspace model and only uses the reduced set of labeled training data for SVM-training.

The recognition rates that we report represent the recall of the classifier with respect to an activity, i.e., the number of correctly classified samples divided by the total number of samples for a given activity.

5.4.1 Data Set and Features

For our experiments we use a dataset published by Kern et al. [Kern *et al.* 2003]. It consists of eight everyday activities, namely *sitting, standing, walking, walking upstairs, walking downstairs, shaking hands, writing on the whiteboard* and *typing on a keyboard*. The activities were recorded by twelve 3D acceleration sensor nodes distributed over the

user's body. The sensors were attached to the ankles, knees, elbows, shoulders, wrists and to both sides of the user's hip. Each node consists of two 2D-accelerometers fixed at an angle of 90 degrees. The overall length of the dataset is 18.7 min, recorded at 92Hz.

The data was recorded in one consecutive run in a semi-naturalistic setting, and therefore not each activity is represented by the same amount of data. Thus, in order to avoid bias in our recognition experiments, we use an equal but random amount of data from each activity for constructing our test- and training sets.

The data samples are vectors of 48 acceleration values, from which we compute the running mean and variance over a window of 50 samples (i.e. about 0.5 seconds), which results in a 96-dimensional feature vector. The window is shifted over the data one sample at a time. In general one could use smaller overlaps between windows, but we found that smaller overlaps have a negative impact especially for small amounts of training data. Since our experiments focus on reducing the amount of training data, we decided to use the maximum overlap for best performance. We did not, however, use individual features and window lengths for each activity, since our goal was not to maximize the classification performance for specific activities.

Mean and variance of the acceleration signal are cheap to compute and have successfully been used for recognizing the activities we are considering (e.g. [Bao and Intille 2004, Kern *et al.* 2003, Krause *et al.* 2003, Ravi *et al.* 2005]). We use these simple features for all approaches described in this chapter.

5.4.2 Naïve Bayes

We use a naïve Bayes classifier as a baseline for our experiments. It is a generative, supervised approach which requires labeled training data for classification. Despite its simplicity, naïve Bayes has yielded high recognition rates for the activities and features we use (e.g. [Kern *et al.* 2003, Van Laerhoven *et al.* 2003]).

Bayes' rule states that the probability $p(a|\mathbf{x})$ of an activity a given an n-dimensional feature vector $\mathbf{x} = x_1, ...x_n$ can be calculated as

$$p(a|\mathbf{x}) = \frac{p(\mathbf{x}|a)p(a)}{p(\mathbf{x})}.$$

In this equation, $p(a)$ denotes the a-priori probability of the activity. The a-priori probability $p(\mathbf{x})$ of the data is only used for normalization. We ignore it in our experiments, since we are only interested in relative likelihoods and not absolute probabilities.

Assuming that the different components x_i of the feature vector $\mathbf{x}$ are independent, we can compute the likelihood $p(\mathbf{x}|a) = \prod_{i=1}^{n} p(x_i|a)$ from labeled training data.

Experiments. We represent each probability density function $p(x_i|a)$ by a 100 bin histogram. As stated, we use 96-dimensional vectors of running mean and variance over a window of 50 samples as features. We performed five experiments, each time reducing the amount of training data by a factor of two. We started with 80% training data and 20% test data, and gradually decreased the amount of training data to 5%, while leaving the amount of test data fixed. The amount of test data stayed constant for all experiments. Each experiment was repeated five times with different parts of the data for training and testing, and the average over all runs was taken as result. For the initial experiment (80% training, 20% test data), this corresponds to a standard 5-fold crossvalidation.

Results. Table 5.1 shows the results of the experiment. Recognition of the activities obviously strongly depends on the amount of available training data. When looking at the average recognition rate, one observes that it steadily drops by about 6% each time the amount of training data is halved, from 73.5% in the beginning to 50.1% percent in the end. This reduction in overall performance is to be expected and a main motivation to search for methods that can obtain high recognition rates using small amounts of training data.

Also, one can observe that the recognition rates vary greatly between the different activities. Stationary activities such as *sitting* and *standing* achieve higher rates than dynamic activities such as *walking upstairs* or downstairs. The highest rate is achieved for *standing* (95%), and the lowest for *walking upstairs* (51.4%). When comparing the different configurations, the rates for *standing* and *sitting* stay relatively stable as the amount of training data is reduced. Presumably this is because there is not much variation in the features for these activities, which means that a small amount of samples is already enough to capture the characteristics of the activity. *Writing on a whiteboard* (which is similar to *standing*) has also high rates, but eventually drops from 93.6% in the beginning to 81.3% when only 5% of the data are used for training. The rates for *walking* drop significantly – from 86.5% to 40.8% – when the training data is reduced, as do the rates for *shaking hands* (from 53.7% to 9.8%). The rates for *walking upstairs* and *walking downstairs* also both drop by about 30%.

5.4.3 Multiple Eigenspaces

For this experiment we first trained the multiple eigenspace algorithm on the unlabeled features, i.e. on the mean and variance computed over a sliding window of 50 samples. In order to reduce the time and space complexity of the growing and selection phases of the algorithm, we initially performed a k-means clustering ($k = 100$)[1] on the features and used the resulting cluster centers as seeds for the growing phase.

[1] Using other cluster numbers than 100 only had small effects on recognition performance, thus we only report on the results for $k = 100$.

	Amount of Training Data				
Activity	80%	40%	20%	10%	5%
stand	95.0	94.8	94.5	94.1	93.5
sit	91.7	92.3	92.8	91.6	92.5
walk	86.5	79.9	70.6	55.6	40.8
upstairs	59.3	50.4	39.1	36.3	31.2
downstairs	51.4	35.4	26.0	23.2	19.9
shake hands	53.7	43.9	32.5	20.4	9.8
whiteboard	93.6	93.2	91.4	87.1	81.3
keyboard	57.0	50.8	56.0	45.6	31.8
Average	73.5	67.6	62.9	56.7	50.1

***Table 5.1**: Recognition Rates using naïve Bayes, for different amounts of training data. The amount of test data was left fixed at 20%.*

Next we assigned activities to the resulting models (i.e. eigenspaces). First we assigned to each training sample the model with the lowest reconstruction error. Then, using the labels of the training samples, we counted which activity was associated most often with a given model. This activity was then assigned to the model, so that the model could be used later for classifcation. Again, we conducted five experiments in total, each time reducing the number of labeled samples used for finding the mapping between models and activities. We decreased the amount of labeled samples in the same way as for the previous experiment (from 80% to 5% of the entire dataset) and left the size of the test set unchanged.

For testing, we assigned to each test sample the activity of the model with the lowest reconstruction error. Each configuration was run five times with different parts of the data for training and testing, and the average over all five runs was taken as result.

Results. Table 5.2 shows the results of the experiment. The distribution of rates differs in various aspects from result of the naïve Bayes experiment. When using 80% training data, the average recognition rate is slightly lower than the one of the naïve Bayes experiment. However, the strength of the approach becomes visible when looking at the runs with reduced training labels – different from the supervised approach, the average rate does not drop but consistently stays at around 70%. This tendency is visible for all individual activities – for none of them, the recognition rate drops by more than 6% when reducing the labeled training set from 80% to 5% of the dataset. Throughout all configurations, *standing*, *writing on a whiteboard* and *typing on a keyboard* achieve the highest recognition rates, with *typing on a keyboard* consistently scoring over 90%.

	Amount of Training Data				
Activity	80%	40%	20%	10%	5%
stand	82.4	85.8	84.1	84.3	82.2
sit	43.1	43.4	52.1	54.8	47.1
walk	74.9	74.6	75.4	74.2	73.1
upstairs	68.7	67.8	68.6	64.0	64.0
downstairs	59.2	63.2	62.8	59.8	53.3
shake hands	52.6	53.0	59.5	53.9	53.6
whiteboard	81.8	78.6	77.2	78.9	78.8
keyboard	92.1	90.2	90.7	91.0	91.8
Average	69.4	69.6	71.3	70.1	68.0

***Table 5.2**: Recognition Rates using Multiple Eigenspaces, for different amounts of training data. The amount of test data was left fixed at 20%.*

5.4.4 Multiple Eigenspaces combined with SVMs

In this experiment we first trained the multiple eigenspace algorithm in the same way as in the previous experiment. Again, the seeding of the eigenspace growing was performed using kmeans clustering ($k = 100$) on the training features. After obtaining the eigenspaces, we trained an SVM with features constructed from the reconstruction errors, as described in Section 5.3.2.

As in the previous experiments, each configuration of test and training sets was run five times, and the average over all five runs was taken as result.

Results. Table 5.3 shows the results of this third experiment. When using all annotations and as expected from discriminant training of the SVMs, the recognition rates are significantly increased w.r.t. to both other approaches. The average performance is more than 88% with most rates above 85% and a maximum of 98% for *typing on a keyboard*. For example, *walking upstairs* and *walking downstairs* both reach over 90%, which is considerably higher than in both of the previous experiments. *Shaking hands* also has rates which are 30% to 40% higher than in the previous experiments. As the number of annotations is reduced, the average recognition rate drops from 88.8% to 64.7%, which is a similar dropoff compared to the naïve Bayes approach. However, the absolute rates are about 15-20% above those of the naïve Bayes experiment.

5.4.5 Discussion

Figure 5.3 shows the performance of all three approaches in one plot. The plot shows the average recognition rate as the size of the labeled training data is successively reduced by

	Amount of Training Data				
Activity	80%	40%	20%	10%	5%
stand	88.0	82.3	80.6	75.5	66.2
sit	83.2	78.6	73.8	69.2	48.6
walk	79.7	68.8	71.1	60.9	51.1
upstairs	91.7	89.7	82.1	78.8	74.9
downstairs	95.0	92.1	88.3	75.2	78.8
shake hands	85.6	85.2	75.9	72.1	67.3
whiteboard	85.8	76.7	75.6	59.9	50.5
keyboard	98.0	94.1	94.9	87.3	80.2
Average	88.3	83.4	80.3	72.4	64.7

***Table 5.3**: Recognition Rates using Multiple Eigenspaces combined with an SVM, for different amounts of training data. The amount of test data was left fixed at 20%.*

a factor of two. On this exponential scale, the performance of the supervised naïve Bayes approach decreases almost linearly. In contrast, the unsupervised multiple eigenspace approach has an almost constant performance throughout all configurations. While starting slightly lower (69% compared to 73%) than naïve Bayes, it outperforms the supervised approach already when the labeled training data is cut in half. The recognition performance of multiple eigenspaces stays stable until the last configuration, in which the labels are reduced to one sixteenth of the original amount. This clearly shows the advantages of this unsupervised and generative approach. The information it extracts from the unlabeled training data helps it to maintain its classification performance as the amount of labeled data is reduced. More importantly, this implies that the multiple eigenspace approach can help to reduce the amount of supervision – and thus the amount of manual work by users – which is required for activity recognition. In contrast, since the supervised approach can only learn from labeled data, it is strongly dependent on annotation, as can be seen from the performance drop in Figure 5.3.

The third curve of the plots in figure 5.3 shows the performance of the integrated approach. Clearly, the overall performance when using 80% training data is far above the performance of the two other approaches. Even though the performance drops as the amount of training data is reduced, the performance using 20% of training data is still clearly above the performances of both approaches. Only when the amount of training data is further reduced to 10% the performance gain using SVM training becomes negligible. When using only 5% of the training data the performance stays behind the multiple eigenspace approach as the SVM training aims to generalize from the rather limited set of training samples.

Overall we can make three observations. First, the use of multiple eigenspaces can reduce the effort of supervision to very small amounts of training data while still preserving a good and constant level of recognition performance. Second, the integration of

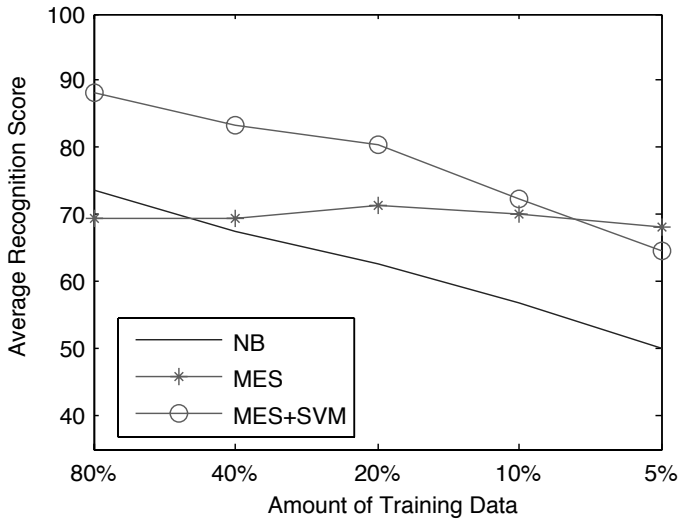


***Figure 5.3**: Comparison of Recognition Performance of the three approaches (Naïve Bayes, Multiple Eigenspaces, and Multiple Eigenspace combined with Support Vector Machines)*

SVM-training on subsets of the training data can increase the overall recognition performance substantially assuming a sufficient amount of labeling data. Third, when labeling and amount of supervision are further reduced, the discriminant nature of SVM learning may not help anymore or can even hurt recognition performance.

5.5 Conclusion

This chapter has introduced an integrated approach combining the advantages of generative modeling and discriminant learning. More specifically the generative approach of multiple eigenspaces was used to obtain a low-dimensional representation of sensor data in a fully unsupervised fashion. In particular the approach allows to model effectively different activities without prior training, user annotation, or any information about the number of activities involved. Support vectors machines are then trained on labeled subsets of the training data to boost the recognition accuracy of the purely unsupervised approach of multiple eigenspaces.

Our experiments yielded three important results. First, the experiments showed that the multiple eigenspace approach can achieve a comparable performance to a baseline system using naïve Bayes classification. Second, we showed that the performance of the multiple eigenspace approach remains high even when the amount of supervision is reduced substantially from 80% to only 5%. Third, the experiments showed that the combined approach does indeed increase recognition performance substantially w.r.t. both the purely unsupervised approach of multiple eigenspaces and the baseline recognition system based on naïve Bayes.

Interestingly, the experiments and the discussion of the previous section also suggest that neither the multiple eigenspace approach nor the discriminant learning approach are sufficient. While the generative eigenspace approach obtains a constant performance even in the presence of decreased supervision, the discriminant learning using SVMs clearly obtains the best recognition performance when enough training data is available. However, when the amount of training data is reduced the performance gain is negligible and might be even reversed. In our opinion, these observations support the claim that in order to obtain scalable activity recognition for real world scenarios we should aim to optimally combine generative with discriminant models.

6

Towards Recognition of High-Level Activities

High-level and longer-term activity recognition has great potentials in areas such as medical diagnosis and human behavior modeling. However, current research in activity recognition mostly focuses on low-level and short-term activities. In this chapter we make a first step towards recognition of high-level activities as they occur in daily life. We use a realistic 10h data set to analyze the performance of four different algorithms for the recognition of both low- and high-level activities. Here we focus on simple features and computationally efficient algorithms as this facilitates the embedding and deployment of the approach in real-world scenarios. While preliminary, the experimental results suggest that the recognition of high-level activities can be achieved with the same algorithms as the recognition of low-level activities.

6.1 Introduction

There are various reasons why only a few researchers have worked on longer-term, complex and high-level activities (with some notable exceptions, as discussed in Section 2.4.1). For example, it is often argued that the recognition of low-level activities is a prerequisite to recognize more complex and high-level activities. Besides being tedious and time-consuming, the recording of high-level activities is a non-trivial task, as the data should be as realistic and representative as possible. Thus, fundamental problems such as the inherent difficulties and the large variability as well as more practical reasons seem to have prevented most researchers to address the recognition of complex and high-level activities.

The explicit goal of the research presented in this chapter is to enable the recognition of longer-term and high-level activities. Therefore, an essential first step for us was to record an interesting and realistic dataset of high-level activities. In this chapter we describe this dataset and compare four algorithms both for the recognition of low-level activities as well as high-level activities. For each of the algorithms, we analyze and discuss different parameters such as feature length and sensor placement. The results suggest

that the recognition of high-level activities may be achievable with the same algorithms as for low-level activities. In particular, our results indicate that recognition of high-level activities can be achieved using features computed from raw sensor data alone, without building up any intermediate representation such as a grammar of low-level activities.

The main contributions presented in this chapter are as follows. First, the results of our experiments suggest that today's activity recognition algorithms are quite capable to address the problem of high-level activity recognition. Second, we record and provide an interesting and realistic dataset of high-level activities which is available from the authors on request. Third, we analyze and compare different algorithms for the recognition of low-level and high-level activities. Fourth, we systematically analyze important parameters such as sensor placement, feature length and classification window.

The chapter is structured as follows. In the next section we introduce the dataset that we recorded and the hardware we used for our experiments. Section 6.3 presents the algorithms we use for recognition of both high- and low-level activities. Sections 6.4 and 6.5 report on the results for low- and high-level activities, respectively. Section 6.6 presents the summary and conclusion.

6.2 Experimental Setup

An important first step towards the recognition of high-level activities is a realistic and representative recording of sensor-data. To this end we formulated four requirements and considerations as the basis of our data recording. First, as the primary aim was the recognition of high-level activities, we explicitly started with the recording of such activities and later defined, named and annotated low-level activities that were performed during these high-level activities. As we will see below, this leads to quite a different set of low-level activities than one may obtain when starting from low-level activities. Second, the recording should be as realistic as possible so that the activities should be performed "in the field" – that is in an unconstrained and natural setting – and not in a laboratory or staged setting. Third, the usefulness and the usability of high-level activity recognition strongly depends on the price and form-factor of the final device. Therefore we decided to keep the algorithms, features and the sensor-platform as simple and power-efficient as possible so that the embedding into a simple self-contained device is feasible in the future. Forth, we decided to start with the recording of data for a single user, as our primary aim was to analyze and show the feasibility of high-level activity recognition first. Even though that might seem like a limitation, we rather expect that the execution of high-level activities varies greatly between individuals so that one might need to use a personalized device.

One requirement formulated above was to base our recognition on simple sensors and easy-to-compute features which is why we decided to use the mean and variance of acceleration signals. Accelerometers are especially appealing in this context, since they are cheap and can be increasingly found in everyday objects such as mobile phones,

cameras, wrist watches and even shoes. The use of simple features for recognition would allow the computation to take place online on a miniature mobile device without draining the battery or slowing down other applications. Computing the features on the device and discarding the raw signals can also help to save memory and allow for longer recordings.

6.2.1 Dataset

During the recordings the user was wearing three sensors. One sensor was attached to the right wrist, one to the righthand side of the hip, and one to the right thigh, as illustrated in Figure 6.3(a). The ground truth labels were mainly added and edited offline, using a separate video recording (from a passively mounted video-camera used during the *housework* and *morning* scenes) and some optional online annotations from a PDA.

The dataset consists of three different high-level activities or *scenes* performed by one user. The first scene consists of a typical morning routine one might perform before going to work, which, for one of the recordings, looked as follows (see Figure 6.1 for the corresponding ground truth annotation): after some time of sleeping, the user gets up, walks to the bathroom, uses the toilet and brushes his teeth. After having breakfast, he leaves the house and drives to work by car. The second scene is a shopping scenario which might look as follows: after working at the computer for some time, the user walks to his car and drives to a nearby shopping center, buys groceries and heads back in his car. In the third scene, the user does some housework after getting up. He might first brush his teeth and have some breakfast, may then wash the dishes, vacuum his apartment and iron some clothes, and eventually walk out of the house.

Each scene was recorded four times, on different days and in a natural environment, i.e. at the user's home and in a nearby supermarket. The scenes were loosely defined by the fact that each activity should at least occur once in each instance. The length of the scenes varies between 40 and 80 minutes; the total length of the data is 621 minutes. Figure 6.1 shows the ground truth for one instance of each scene, and Figure 6.2 gives an overview of all activities. The scenes consist of 15 different activities (plus one garbage class for unlabeled data), some of which are shared between two or three scenes. For evaluation, we created four sets, each consisting of three concatenated scenes. We used these sets to perform a 4-fold leave-one-out crossvalidation on the data.

6.2.2 Hardware

Figure 6.3(b) shows the sensor platform that was used for recording the data for our experiments [Van Laerhoven *et al.* 2006]. It features a 2D accelerometer (ADXL202JE) and nine binary tilt switches for sensing motion and orientation of the user. Besides giving a coarse sense of orientation, the tilt switches can be used to control the system's transition to and from a low-power mode. In low-power mode, the accelerometer is switched off and the frequency of the micro controller (16F628A) reduced from 4 MHz to 48 kHz,

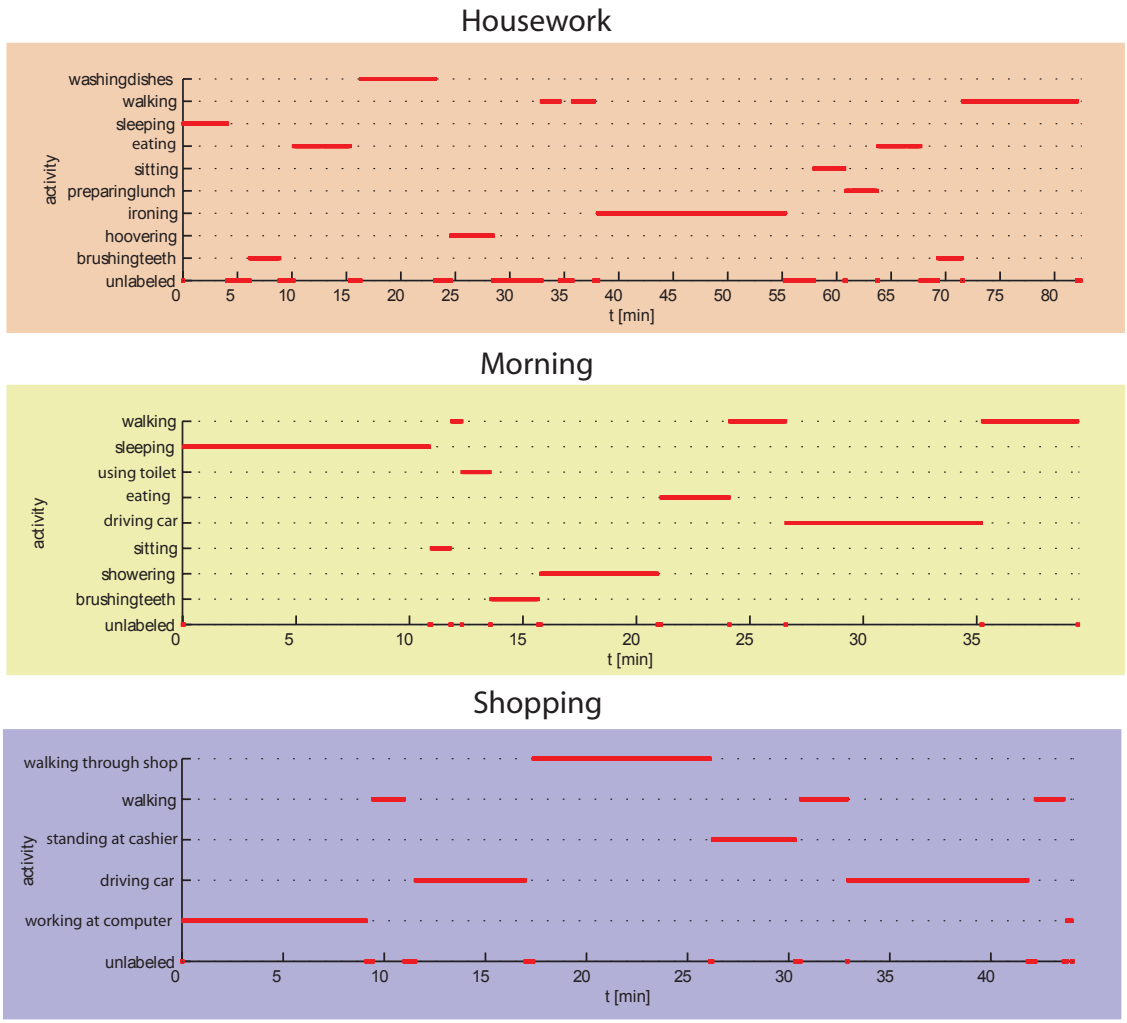


***Figure 6.1**: Ground truth for recordings of the three scenes* Housework, Morning *and* Shopping. *Each scene was performed four times by the user, here we show only one instance of each scene.*

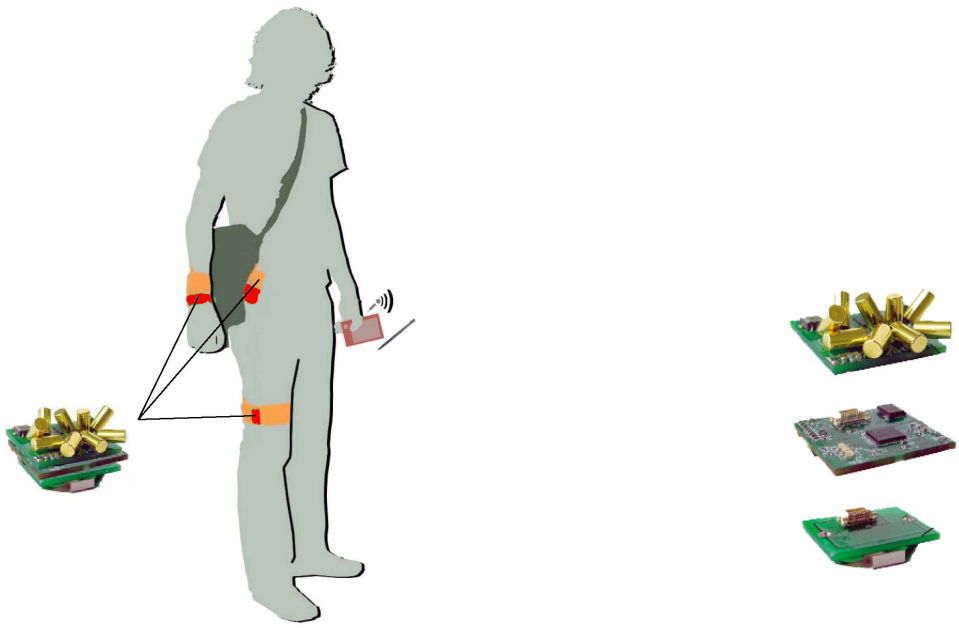
which can extend the battery lifetime of the platform from about one day to one month. The sensor board is stacked onto a BSN node [Lo *et al.* 2005] with 512 kb of EEPROM storage for logging sensor data, followed by a third board for the power supply.

6.2.3 Feature Computation

During recordings, the platform stores all sensor data on the EEPROM storage, from which it can later be retrieved via an rs232 connection. We aimed for continuous recordings of several hours, and the limiting factor for our experiments was the size of the 512 kb on-board memory rather than battery lifetime. To save memory, we compute and store only the mean and variance of the acceleration signal at 2 Hz and discard the raw (80

Highlevel Activities	*Lowlevel Activities*	
a Preparing for Work	1 (unlabeled)	9 walking [a, b]
b Going Shopping	2 brushing teeth [a, c]	10 working at computer [b]
c Doing Housework	3 taking a shower [a]	11 waiting in line in a shop [b]
	4 sitting [a]	12 strolling through a shop [b]
	5 driving car [a, b]	13 hoovering [c]
	6 eating at table [a,c]	14 ironing [c]
	7 using the toilet [a]	15 preparing lunch [c]
	8 sleeping [a]	16 washing the dishes [c]

***Figure 6.2**: Overview of the low- and high-level activities in the recorded dataset. Each high-level activity consists of a set of low-level activities, as indicated in brackets.*



(a) (b)

***Figure 6.3**: Left: User wearing sensors on wrist, hip and thigh. Right: The sensor platform, consisting of the power supply (bottom), the BSN node for logging (middle) and the sensor board (top).*

Hz) acceleration data. This allows us to record about five hours of sensor data on the chip. (The current generation of the platform has a larger on-board memory and allows for continuous recordings of several days or even weeks.)

6.3 Algorithms

We use four different approaches for recognition of activities – three of them are based on a discrete representation that we obtain by clustering the sensor data, and one approach is based on training HMMs on continuous data. All approaches have in common that they use the mean and variance of the acceleration signal over a sliding window as the underlying features. These features are cheap to compute and are known to yield high recognition rates in settings comparable to ours (e.g. [Bao and Intille 2004, Kern 2005, Ravi *et al.* 2005, Huỳnh and Schiele 2006a] (see also Chapter 5)).

Related work has shown that it is possible to recognize movements or activities based on low dimensional models learned in a semi- or unsupervised fashion (e.g., [Minnen *et al.* 2006b, Huỳnh and Schiele 2006a] (see also Chapter 5)). Such models can also be thought of as an alphabet of symbols, a vocabulary in which activities are formulated as 'sentences'. Compositions of such sentences could later serve as a tool for recognizing more abstract and high-level behavior. The first three of the following approaches is inspired by this idea, but as we do not assume that human motion follows a strict grammar, we only consider histograms of symbols over intervals, without modeling their temporal order. We use k-means clustering as a simple yet effective unsupervised method to map features to a set of discrete symbols, i.e. to one of the k cluster centroids. We represent each feature by the closest cluster centroid. As a result, the input data is transformed into a one-dimensional sequence of cluster assignments. Based on this representation, we employ three different learning methods which we describe in the following. The fourth method is based on HMMs and uses a vector of mean and variance values as features. Figure 6.4 gives a conceptual overview of the different representations used for recognition, and Figure 6.5 shows a real-world example. In the following we briefly outline the four approaches we used.

Clustering + NN. As a baseline method, we cluster the training samples using k-means and label each cluster with the activity that occurs most often among the training samples belonging to the cluster. Classification is then performed by a nearest neighbor (NN) classification using the cluster centroids, i.e. we assign to each test sample the label of the closest cluster centroid. During experiments we vary the size of k and the length of the window over which the features are computed.

Histograms + NN. In this approach, rather than using individual symbols as features, we compute histograms of cluster assignments over a sliding window of the training sequence. Each histogram is labeled with the activity that occurs most often in the window of samples that it covers. For evaluation, we perform a nearest neighbor (NN) classification on the histograms computed from a test sequence.

Histograms + SVM. This approach is also based on histograms of cluster assignments. However, instead of using a nearest neighbor classifier, we train a support vector machine

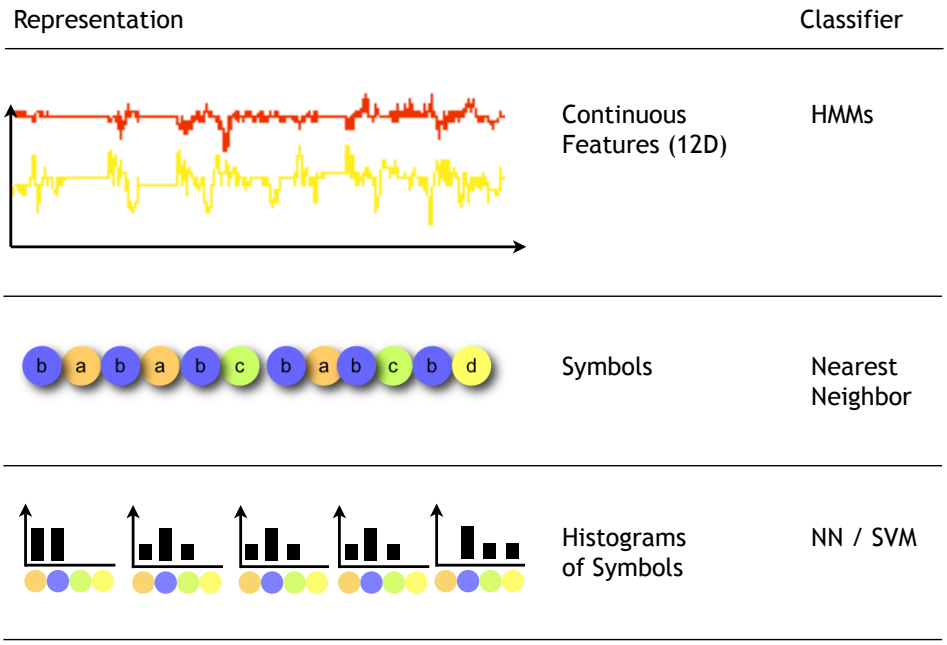


***Figure 6.4**: Conceptual overview of the representations and classifiers we used for recognizing both high-level and low-level activities. The symbols are obtained by clustering the continuous features – they basically correspond to clusters, each represented by the cluster centroid. The histograms are computed from a sliding window over a stream of symbols.*

(SVM) using the histograms as features.

HMMs. The fourth approach is based on Hidden Markov Models (HMMs). HMMs belong to the class of generative statistical signal models, and they have been successfully used in activity recognition tasks before (e.g. [Oliver *et al.* 2002, Clarkson and Pentland 1999, Lester *et al.* 2005, Lukowicz *et al.* 2004]). They lend themselves to a hierarchical classifier design, which makes them interesting candidates for modelling activities on different levels of abstraction. For this approach we also use the mean and variance of the acceleration signal over a sliding window as features. We then partition the data into N equal parts and train a separate HMM on each part. We use left-right models with one Gaussian per state, and we vary the number of states in our experiments. In order to assign activity labels to the models, we use a sliding window over the features as observation sequence, and compute the likelihood of the window for each of the N models. The model with the highest likelihood is then assigned the label of the activity that occurs most often in the window. Classification is performed similarly, i.e. by computing the likelihood of each model over a sliding window starting at a certain sample, and subsequently assigning to the sample the label of the model with the highest likelihood.

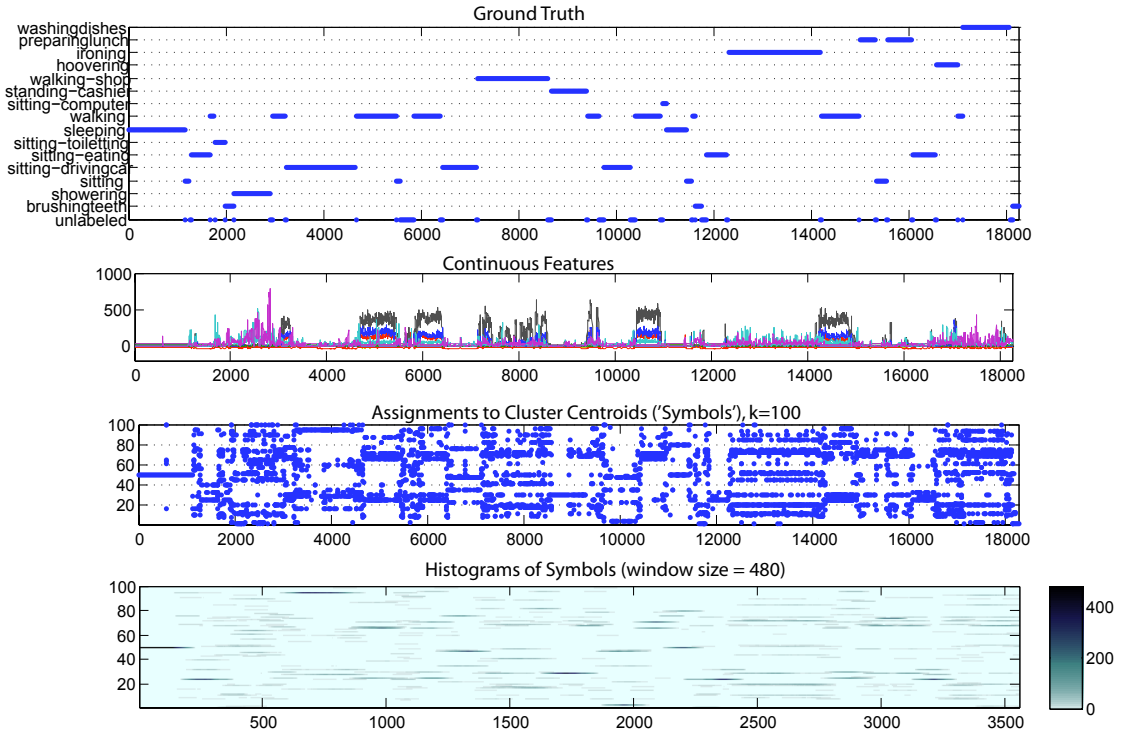
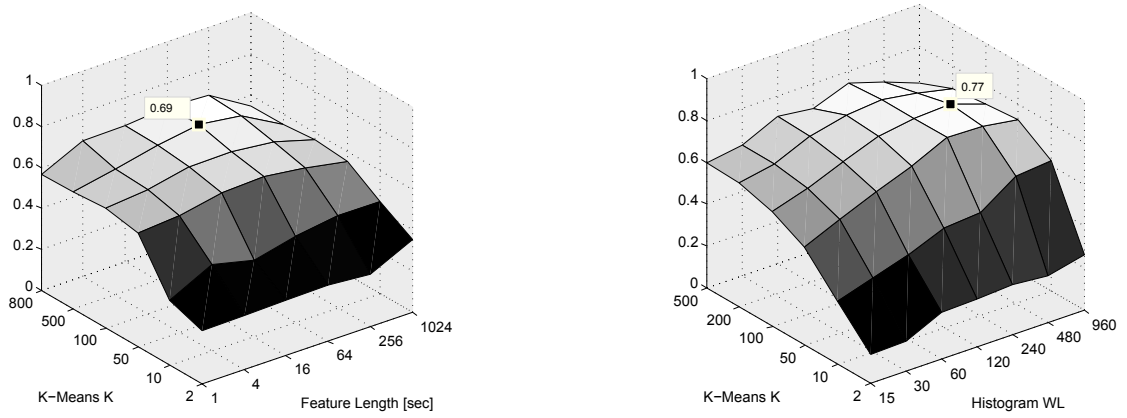


Figure 6.5*: Example of the different representations used for recognition. From top to bottom: ground truth; features (mean & variance over 4 sec); cluster assignments (each feature is assigned to one of k=100 clusters); histograms of cluster assignments (over windows of 480 samples).*

6.4 Low-level Activities

In this section we report on the performance of the above mentioned approaches with respect to the fifteen low-level activities listed in Figure 6.2. As mentioned earlier, we defined the low-level activities after the recording of the high-level activities. That way, a somewhat obvious but important observation is that the definition of low-level activities is not as well-defined as one might expect. E.g., for the following activities, it is not clear if they belong to the same or to different low-level activities: *walking down a corridor* vs. *walking in a supermarket while collecting items*; *sitting in a car* vs. *sitting at a table while eating* vs. *sitting on the toilet* vs. *sitting at a desk and working on a computer*; etc. It should be clear that this is not simply a question of a hierarchical and temporal decomposition of concurrent activities, but rather an inherent difficulty linked to the context of the particular activity (e.g. *sitting on the toilet* vs. *sitting at a table*). As a consequence, we decided to define the low-level activities within each high-level activity as they occurred within the context of the high-level activity. That way we have a range of activities which occur across multiple high-level activities such as *walking, eating at table* and *brushing teeth* and others which are more specific such as *driving a car* or *strolling through a shop*.

Based on these definitions of low-level activities, this section compares the recogni-



(a) (b)

***Figure 6.6**: Accuracy of classification for low-level activities; using assignments to cluster centroids as features (left) vs. using histograms of such assignments in combination with nearest neighbor classification (right).*

tion performance of our four approaches. For each of the algorithms we also identify and discuss suitable parameters such as the number of clusters, the length of the feature window, and also appropriate on-body locations for the sensors.

Clustering + NN. Figure 6.6(a) shows the accuracy[1] for different numbers k of clusters and different window lengths for the features. One can observe that values of k below 50 have a negative impact on the recognition performance. For values of $k >= 50$, accuracy lies roughly between 60% and 70%. The best result of 69,4% is obtained for $k = 500$ and a feature length of 64 seconds. Surprisingly, the best results are obtained for relatively long window lengths. Lengths between 16 and 256 seconds perform best, and there is a visible drop in performance for shorter and longer window lengths. Figure 6.7 shows the confusion matrix for the best parameter combination. One can clearly see that the recognition performance varies strongly between the different activities. Seven of the 15 activities have recall or precision values above 70%, the best being *sleeping* (97.4/90.6), *working at the computer* (89.9/78.5), *walking* (82.1/78.4) and *driving car* (79.7/88.8). During four activities the user was sitting (*sitting, driving car, eating at table, using the toilet*), and from Figure 6.7 one can see that these activities are often confused with each other during classification.

Histograms + NN. Figure 6.6(b) shows the recognition results for the histogram-based approach combined with a nearest neighbor classifier. We vary the number of clusters and the length of the histogram windows (the windows are always shifted by 5 features at a time). The underlying mean and variance features are computed over windows of 4

[1] we use the term *accuracy* to refer to the number of correctly classified samples divided by the number of all samples

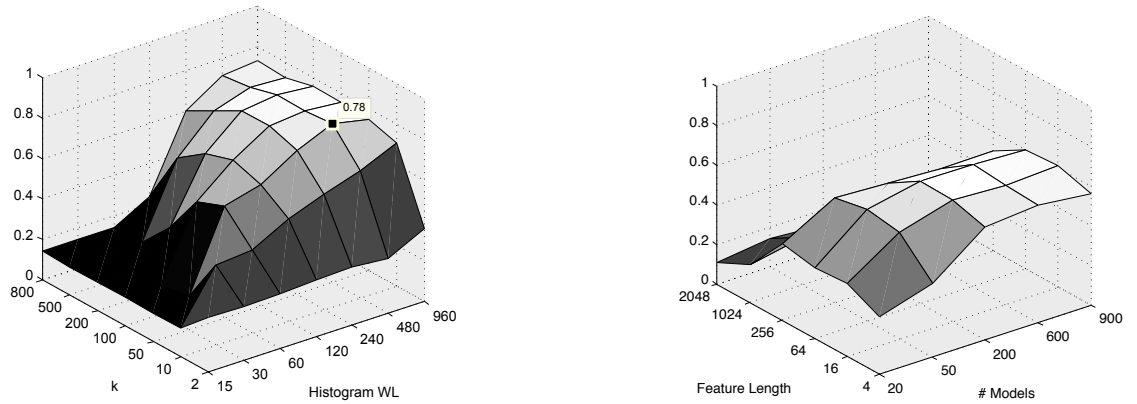
Ground Truth	Classified Activity 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Sum	Recall
1 unlabeled	**1314**	148	188	64	567	136	81	103	540	85	10	428	317	993	707	158	5839	**22.5%**
2 brush teeth	146	**1258**	310	0	0	0	0	0	9	0	16	0	71	302	2	51	2165	**58.1%**
3 shower	83	249	**1710**	0	0	13	17	0	58	0	0	47	54	270	40	70	2611	**65.5%**
4 sit	287	4	6	**684**	424	168	267	5	68	19	0	49	7	26	19	0	2033	**33.6%**
5 drive car	334	0	0	343	**9743**	301	84	26	62	0	0	73	0	0	0	0	10966	**88.8%**
6 eat	192	5	21	127	938	**4253**	26	11	42	0	0	25	2	3	21	13	5679	**74.9%**
7 use toilet	83	14	46	41	324	106	**224**	7	14	16	0	0	12	53	4	0	944	**23.7%**
8 sleep	260	14	21	45	116	34	0	**7016**	111	36	0	55	0	0	10	22	7740	**90.6%**
9 walk	614	12	105	29	66	0	7	0	**8988**	0	0	1285	139	153	32	40	11470	**78.4%**
10 work at comp.	99	0	3	22	52	35	15	36	26	**1325**	0	24	9	21	21	0	1688	**78.5%**
11 stand at cashier	14	0	0	0	0	0	0	0	0	0	**798**	717	23	145	92	15	1804	**44.2%**
12 walk in shop	193	14	37	7	2	0	0	0	836	0	297	**3260**	109	201	342	14	5312	**61.4%**
13 hoover	74	44	74	0	0	0	0	0	128	0	0	135	**785**	456	66	149	1911	**41.1%**
14 iron	122	76	155	0	0	0	0	0	38	0	162	53	267	**7009**	438	263	8583	**81.7%**
15 prep. lunch	331	4	2	0	0	20	0	0	14	0	37	349	49	499	**731**	95	2131	**34.3%**
16 wash dishes	240	29	54	13	0	0	0	0	11	0	3	37	23	350	255	**2554**	3569	**71.6%**
Sum	4386	1871	2732	1375	12232	5066	721	7204	10945	1481	1323	6537	1867	10481	2780	3444	74445	
Precision	**30.0%**	**67.2%**	**62.6%**	**49.7%**	**79.7%**	**84.0%**	**31.1%**	**97.4%**	**82.1%**	**89.5%**	**60.3%**	**49.9%**	**42.0%**	**66.9%**	**26.3%**	**74.2%**		

***Figure 6.7**: Aggregate confusion matrix for the best parameter combination when using cluster centroids as features. k = 500, mean & var computed over 64 seconds, shift = 0.5 seconds. Overall accuracy is 69%.*

Ground Truth	Classified Activity 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Sum	Recall
1 unlabeled	**79**	79	24	0	7	4	34	50	53	0	0	0	10	208	11	66	625	**12.6%**
2 brush teeth	73	**142**	86	0	0	8	8	1	0	0	0	3	0	0	0	59	380	**37.4%**
3 shower	0	0	**558**	0	0	0	0	0	0	0	0	0	0	0	0	3	561	**99.5%**
4 sit	0	0	0	0	67	0	68	0	26	0	0	0	0	0	27	0	188	**0.0%**
5 drive car	0	0	0	0	**2134**	0	0	0	102	0	0	54	0	0	0	0	2290	**93.2%**
6 eat	0	10	13	0	104	**1033**	25	0	0	0	0	47	5	0	11	23	1271	**81.3%**
7 use toilet	0	81	5	0	71	28	**26**	10	13	0	0	0	0	0	0	0	234	**11.1%**
8 sleep	15	26	0	0	0	4	7	**1498**	7	0	0	0	0	0	0	0	1557	**96.2%**
9 walk	46	0	6	0	90	0	17	7	**1632**	18	0	303	10	69	0	3	2201	**74.1%**
10 work at comp.	0	0	0	0	127	0	0	0	17	**180**	0	0	0	0	0	0	324	**55.6%**
11 stand at cashier	0	0	0	0	0	0	0	0	3	0	**125**	120	0	0	139	0	387	**32.3%**
12 walk in shop	0	0	0	0	23	0	0	0	52	0	60	**886**	0	0	75	0	1096	**80.8%**
13 hoover	10	0	0	0	0	0	0	0	0	0	0	0	**365**	8	10	21	414	**88.2%**
14 iron	0	0	0	0	0	10	0	0	10	0	0	0	0	**1676**	45	10	1751	**95.7%**
15 prep. lunch	0	0	8	68	0	17	30	0	5	0	30	53	13	14	**156**	109	503	**31.0%**
16 wash dishes	0	0	12	0	0	17	0	0	6	0	0	14	6	0	0	**715**	770	**92.9%**
Sum	223	338	712	68	2623	1121	215	1566	1926	198	215	1480	409	1975	474	1009	14552	
Precision	**35.4%**	**42.0%**	**78.4%**	**0.0%**	**81.4%**	**92.1%**	**12.1%**	**95.7%**	**84.7%**	**90.9%**	**58.1%**	**59.9%**	**89.2%**	**84.9%**	**32.9%**	**70.9%**		

***Figure 6.8**: Aggregate confusion matrix for the best parameter combination when using histograms of symbols (cluster centroids) as features. k = 100, histogram windows over 480 features (about 4 min.) shifted by 5 features each, mean & var computed over 4 sec., shift = 0.5 seconds. Overall accuracy is 77%.*

seconds with a shift of 0.5 seconds (in contrast to the clustering approach, we observed that small feature windows performed better here). The highest accuracy of 77% is obtained for $k = 100$ and a histogram window of 480 samples, covering about 4 minutes of data. For larger histogram windows the accuracy visibly decreases. Similarly to the clustering results, values of k below 50 lead to a sharp drop in performance, implying that too much information is lost from the discretization. Figure 6.8 shows the confusion matrix for the best parameter settings. Except for the activities *taking a shower*, *sitting*, *using the toilet* and *washing the dishes*, the precision increases for all activities compared to the previous approach. Notably, the confusion between the activities *ironing* and *vacuuming* is much lower in this approach. The overall gain in accuracy of 8% indicates that the use of histograms of symbols rather than individual symbols does indeed help to improve recognition performance.



(a) (b)

***Figure 6.9**: Accuracy of classification for low-level activities; using histograms of cluster assignments in combination with an SVM (left) vs. using HMMs (right).*

Histograms + SVM. When using an SVM for classification in combination with the histogram features, the recognition results can be slightly improved compared to the nearest neighbor approach. Figure 6.9(a) shows the accuracy for different values of k and different window lengths for the histograms. The best result of 78% is obtained for $k = 50$ and a histogram window of 480 samples, covering about 4 minutes of data. One can observe that accuracy decreases with higher number of clusters and smaller window lengths. For window lengths between 240 and 960 samples, corresponding to about 2 to 8 minutes of data, and values of k between 50 and 200, we obtain the highest accuracies.

HMMs. Figure 6.9(b) shows recognition results for the HMM approach. We vary the feature length and the number of models N; in this particular example, the number of states is fixed to 8, and the observation window for classification covers 16 samples. The number of models N directly affects the length of data that each HMM models, since the data is equally partitioned into N parts. Thus, N is inversely related to the length of the histogram windows of the previous approaches. From the plot one can observe that using less than 200 models (i.e. each model sees about 2.5 min of data or more) leads to a visible decrease in performance. We obtained the best result of 67% for $N = 200$ models and a feature length of 64 sec, an observation length of 16 and models with 32 states. When varying the number of states we found that they only marginally effected the results. Figure 6.10 shows the confusion matrix for the best parameter combination. Overall, results of the HMM approach suggest that the temporal aspect – at least for the features we employed – is not dominant enough to allow for higher recognition rates.

Sensor placement. The results so far were based on the data of all three sensors the user was wearing on wrist, hip and thigh. It turns out that using only subsets of these sensors for recognition reveals some interesting relations between the placement of sensors and the recognition of individual activities. For instance, we found that the overall

Ground Truth	Classified Activity: 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	Sum	Recall
1 unlabeled	**288**	337	654	849	323	497	533	49	879	458	65	833	103	485	151	98	6602	**4.4%**
2 brush teeth	0	**1565**	32	77	21	32	46	0	20	32	0	0	0	159	183	0	2167	**72.2%**
3 hoover	0	75	**1070**	523	0	33	32	0	120	50	0	0	0	0	0	0	1903	**56.2%**
4 iron	0	0	182	**6732**	1211	41	0	0	117	166	0	0	0	0	130	0	8579	**78.5%**
5 prep. lunch	0	0	0	365	**927**	105	27	0	131	162	66	35	130	33	54	33	2068	**44.8%**
6 sit	10	81	0	61	142	**577**	84	13	273	212	33	257	0	0	0	196	1939	**29.8%**
7 eat	0	0	20	43	70	80	**3615**	0	130	252	0	1327	0	33	19	32	5621	**64.3%**
8 sleep	533	16	0	0	11	283	29	**6638**	56	0	33	127	0	0	0	65	7791	**85.2%**
9 walk	33	130	196	145	7	463	310	0	**8399**	171	0	493	13	903	0	33	11296	**74.4%**
10 wash dishes	0	9	0	167	98	69	206	0	2	**2740**	0	0	0	0	262	0	3553	**77.1%**
11 work at comp.	0	0	0	0	0	65	24	0	178	0	**1307**	33	0	86	0	0	1693	**77.2%**
12 drive car	0	0	0	0	0	99	524	0	440	0	33	**9670**	0	33	0	98	10897	**88.7%**
13 stand at	99	0	0	0	200	0	0	0	0	0	0	0	**1285**	212	0	0	1796	**71.5%**
14 walk in shop	0	98	0	0	197	66	0	0	862	0	0	68	429	**3581**	0	0	5301	**67.6%**
15 shower	0	254	0	205	0	0	0	0	0	130	0	0	0	0	**1982**	0	2571	**77.1%**
16 use toilet	20	16	0	25	0	206	328	0	28	2	0	295	0	0	0	0	920	**0.0%**
Sum	983	2581	2154	9192	3207	2616	5758	6700	11635	4375	1537	13138	1960	5525	2781	555	74697	
Precision	**29.3%**	**60.6%**	**49.7%**	**73.2%**	**28.9%**	**22.1%**	**62.8%**	**99.1%**	**72.2%**	**62.6%**	**85.0%**	**73.6%**	**65.6%**	**64.8%**	**71.3%**	**0.0%**		

***Figure 6.10**: Aggregate confusion matrix for the best parameter combination when using the HMM-based approach. The parameters were: window length for features = 64 sec., 200 models, 32 states per model, observation length = 16. Overall accuracy is 67.4%.*

accuracy of the clustering approach slightly improved from 69% to 70% when we used only two sensors, namely the sensors on wrist and thigh. These results are consistent with the findings from [Bao and Intille 2004], who also found that when using only two sensor locations, wrist and thigh are the most suitable locations. Using these locations even leads to better results when recognizing the activities *brushing teeth*, *driving car*, *preparing lunch* and *washing dishes*. When only using the wrist sensor, performance for *brushing teeth* and *taking a shower* improves, likely because these activities are mainly characterized by hand and arm movements. For *sleeping* and *walking*, using only the hip sensor already yields precision and recall values up to 95%.

6.4.1 Discussion

Figure 6.11 shows a summary table comparing the best results of the four approaches. Generally, the approach *Histograms + SVM* achieves the highest accuracy of 79.1%. For most activities, the use of histograms instead of single cluster assignments as features leads to better precision and recall values. However, there are two stationary (*sitting, using the toilet*) and two dynamic activities (*brushing teeth, walking*) in which the use of single cluster assignments yields higher results in either precision, recall or both. The HMM approach achieves the lowest accuracy of 67.4%, slightly less than the clustering approach. In summary, we conclude that using histograms of symbols as features and combining them with a strong classifier is a promising and competitive approach for recognizing the type of daily activities we recorded in our study.

It is worth noting that the overall recognition scores seem low compared to the published state-of-the-art. However, in contrast to most other recordings and as discussed above, we explicitly defined the low-level activities after the recording of the high-level activities, and therefore both the larger variability within single low-level activities (such as *walking*) and the high similarity between different low-activities (such as *walking* and

	Clusters/NN		Histograms/NN		Histograms/SVM		HMM	
Activity	p	r	p	r	p	r	p	r
(unlabeled)	30.0	**22.5**	**35.4**	12.6	7.9	3.0	29.3	4.4
brush teeth	**67.2**	58.1	42.0	37.4	23.0	21.1	60.6	**72.2**
shower	62.6	65.5	78.4	**99.5**	**86.7**	91.8	49.7	56.2
sit	**49.7**	**33.6**	0.0	0.0	0.0	0.0	73.2	78.5
drive car	79.7	88.8	81.4	93.2	**86.9**	**95.4**	28.9	44.8
eat	84.0	74.9	**92.1**	81.3	82.3	87.3	22.1	29.8
use toilet	**31.1**	**23.7**	12.1	11.1	15.0	9.4	62.8	64.3
sleep	97.4	90.6	95.7	96.2	91.2	**97.2**	**99.1**	85.2
walk	82.1	**78.4**	**84.7**	74.1	79.5	77.6	72.2	74.4
work at computer	89.5	78.5	90.9	55.6	**93.3**	**94.8**	62.6	77.1
stand at cashier	60.3	44.2	58.1	32.3	**75.9**	47.3	85.0	77.2
walk in shop	49.9	61.4	59.9	**80.8**	**70.7**	80.1	73.6	88.7
vacuum	42.0	41.1	89.2	**88.2**	**98.3**	82.6	65.6	**71.5**
iron	66.9	81.7	84.9	95.7	**89.0**	95.9	64.8	67.6
prep. lunch	26.3	34.3	32.9	31.0	**45.7**	54.3	71.3	77.1
wash dishes	74.2	71.6	70.9	**92.9**	**79.0**	89.9	0.0	0.0
Mean	62.0	59.3	63.0	61.4	**64.0**	**64.2**	57.5	60.6
Accuracy	69.4		77.0		**79.1**		67.4	

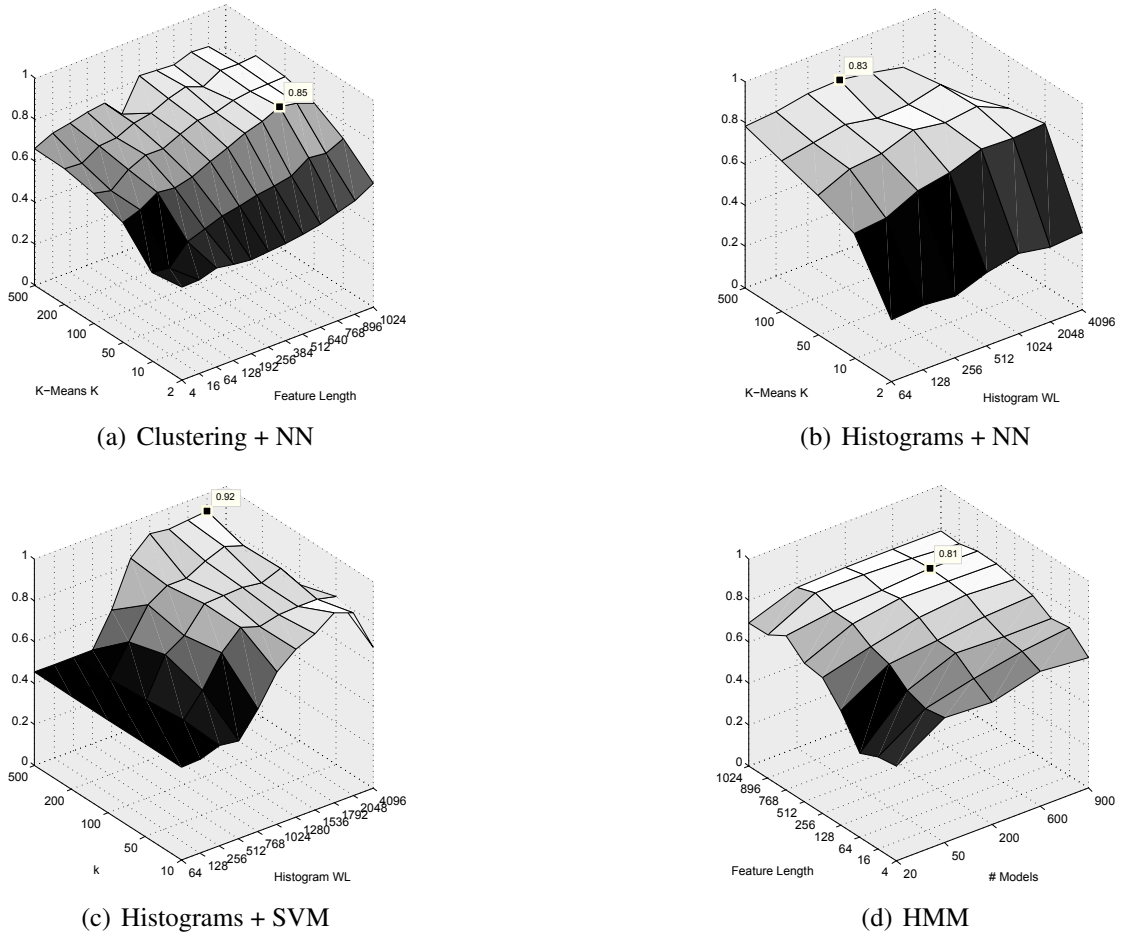
***Figure 6.11**: Summary of the results for low-level activities. Each column shows the precision (p) and recall (r) values for each activity, as well as the accuracy, i.e. the number of correctly classified samples divided by all samples. The highest values in each row are highlighted.*

walking through shop) pose a more challenging recognition problem than is usually addressed.

6.5 High-level Activities

In this section we report on how well our proposed approaches can deal with the recognition of high-level scenes comprising a collection of low-level activities. More specifically, we evaluate how well our algorithms can classify the three different scenes *Morning*, *Housework*, and *Shopping*. Each scene has a length of at least 40 minutes and consists of at least six different activities. The evaluation was performed in the same fashion as for the low-level activities: we constructed four datasets, each containing one instance of each of the three scenes, and then performed a leave-one-out crossvalidation.

Clustering + NN. Figure 6.12(a) shows the accuracy for different numbers of clusters and different window lengths for computing mean and variance of the signal. As for the low-level activities, one can observe that for values of k below 50 performance decreases rapidly. In terms of feature windows, there is a visible tendency that longer window lengths lead to a better performance. For the parameter values that we sampled, the best



(a) Clustering + NN

(b) Histograms + NN

(c) Histograms + SVM

(d) HMM

***Figure 6.12**: Accuracy of classification for high-level activities.*

		Classification (Clusters + NN)					Classification (Histograms + NN)				
		preparing for work	going shopping	doing housework	Sum	Recall	preparing for work	going shopping	doing housework	Sum	Recall
GT	preparing for work	**7652**	921	916	9489	**80.6%**	**1568**	336	72	1976	**79.4%**
GT	going shopping	1030	**6683**	1310	9023	**74.1%**	86	**1669**	101	1856	**89.9%**
GT	doing housework	741	263	**14764**	15768	**93.6%**	354	224	**2651**	3229	**82.1%**
	Sum	9423	7867	16990	34280		2008	2229	2824	7061	
	Precision	**81.2%**	**84.9%**	**86.9%**			**78.1%**	**74.9%**	**93.9%**		

		Classification (Histograms + SVM)			Sum	Recall	Classification (HMM)			Sum	Recall
GT	preparing for work	**1383**	132	0	1515	**91.3%**	**8220**	753	515	9488	**86.6%**
GT	going shopping	62	**1359**	126	1547	**87.8%**	1042	**4962**	930	6934	**71.6%**
GT	doing housework	14	143	**2612**	2769	**94.3%**	1156	536	**7334**	9026	**81.3%**
	Sum	1459	1634	2738	5831		10418	6251	8779	25448	
	Precision	**94.8%**	**83.2%**	**95.4%**			**78.9%**	**79.4%**	**83.5%**		

***Figure 6.13**: Aggregate confusion matrices for the best parameter combinations of the four approaches for recognizing high-level activities.*

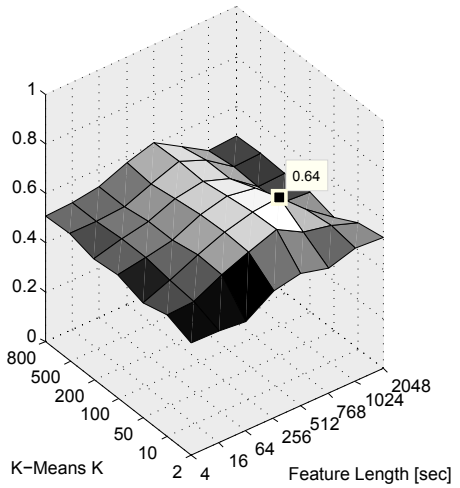
result of 84.9% was obtained for $k = 50$ and a feature window of 768 sec., i.e. about 13 min. (We comment on the feature length below in the paragraph 'Sensor Placement'.) The confusion matrix for this configuration is shown in Figure 6.13 (upper left). Precision and recall range between 74% and 94%.

Histograms + NN. In this experiment, as for the low-level activities, we vary the number of clusters and the length of the histogram. The results can be seen in Figure 6.12(b). The mean and variance features are computed over 4 sec. windows with a shift of 1 second. The best results are obtained for values of k between 50 and 500, and histogram windows between 512 and 2048 samples, i.e. between about 8 and 32 minutes. Figure 6.13 (upper right) shows the confusion matrix for $k = 500$ and a histogram window of 512 samples; the accuracy for this run was 83.4%, which is slightly lower than for the clustering approach. In terms of precision and confusion there is no clear difference to the clustering approach. However, the results improve substantially when using an SVM for classification instead of a nearest neighbor classifier, as is described in the next section.

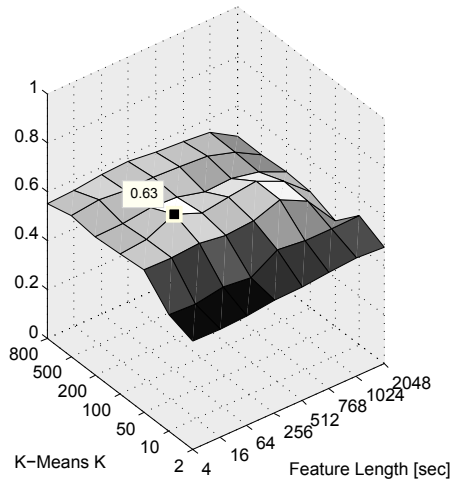
Histograms + SVM. Figure 6.12(c) shows the accuracy for different values of k and different window lengths for the histograms when using an SVM as classifier. The best results are obtained for histogram windows between 1280 and 2048 samples, i.e. between 20 and 32 min. Interestingly, the number of clusters for discretization only has a minimal influence on the recognition performance, the dominating parameter is the length of the histogram window. Even when using only $k = 10$ clusters, the accuracy stays above 90%. Figure 6.13 (lower left) shows the confusion matrix for the best result of 91.8% accuracy, which is an improvement of about 7% compared to using the nearest neighbor classifier as described in the previous paragraph.

HMMs. Figure 6.12(d) shows the recognition results for the HMM approach. As for the low-level activities, we vary the feature length and the number of models N. The number of states is fixed to $s = 2$ (we did vary the number of states but found only small changes in performance), and the length of the observation window for each HMM is set to 16 samples. From the figure one can observe that values of N below 200 lead to a decrease in performance. The best results of slightly above 80% are obtained for feature lengths above 256 seconds (4 min) and $N = 200$ models or more. Figure 6.13 (lower right) shows the confusion matrix for $N = 200$ and a feature length of 768 seconds.

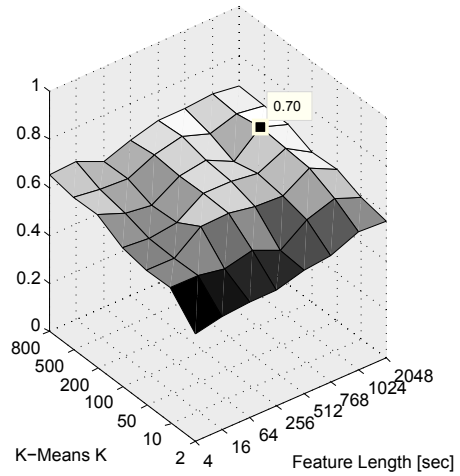
Sensor Placement. We also investigated the influence that different sensor locations have on the recognition of high-level activities. Figure 6.14 shows the differences in performance when applying the clustering approach to subsets of sensors. Figure 6.14(a) shows the results for the wrist sensor. One can observe that for this sensor, the size of the feature window strongly influences the recognition rate – there is a distinct peak for relatively large windows between 512 and 1024 seconds. Obviously, for shorter windows the wrist movements are not discriminative enough for recognition. This might be due to the fact that the three scenes share some of the low-level activities, and that of these, many involve similar wrist movements, as for example *brushing teeth* or *showering*. The results for hip (Figure 6.14(b)) and thigh (Figure 6.14(c)) sensor do not exhibit such a clear tendency towards specific window lengths. Thus it appears that it is mainly the wrist sensor that is responsible for the good performance of relatively long windows when



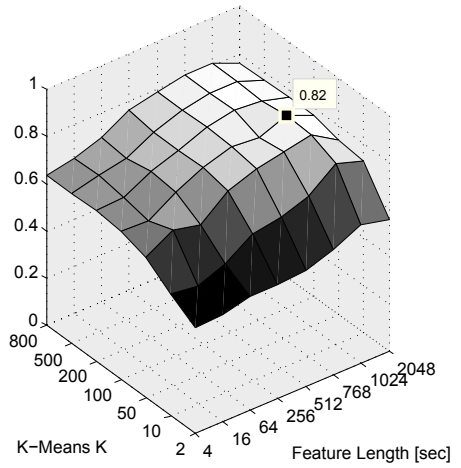
(a) Wrist



(b) Hip



(c) Thigh



(d) Wrist + Thigh

***Figure 6.14**: Clustering + NN - based recognition accuracy of high-level activities for subsets of sensor locations. The best values of each combination are highlighted.*

using all three sensors. The result for the hip sensor indicates that the performance at this location is more influenced by the number of clusters than the feature length; the best results are obtained for $k = 100$. Similarly as for the low-level activities, the combination of wrist and thigh sensor also performs very well for high level activities. For $k = 100$ and a feature length of 1024, the accuracy is 82%, i.e. only 3% worse than when using all three sensors.

	Clusters + NN		*Histograms + NN*		*Histograms + SVM*		*HMM*	
Scene	p	r	p	r	p	r	p	r
Preparing for Work	81.2	80.6	78.1	79.4	**94.4**	**91.3**	78.9	86.6
Going Shopping	84.9	74.1	74.9	89.9	**83.2**	**87.8**	79.4	71.6
Doing Housework	86.9	93.6	93.9	82.1	**95.4**	**94.3**	83.5	81.3
Mean	84.4	82.2	82.3	83.8	**91.1**	**91.2**	80.6	79.3
Accuracy	84.9		83.4		**91.8**		80.6	

***Figure 6.15**: Summary of the results for high-level activities. The columns show the precision (p) and recall (r) values for each activity, as well as the accuracy.*

Parameter	*NN + Clusters*	*NN + Histograms*	*SVM + Histograms*	*HMM*
Window Length of Features	> 10 min	4 sec	4 sec	> 4 sec
# Clusters	>= 50	>= 50	>= 50	
Window Length of Histograms		8-32 min	20-32 min	
# States				2
Observation Length				16

***Figure 6.16**: Parameters that we found worked well for recognizing high-level activities, using our four different approaches.*

6.5.1 Discussion

Figure 6.16 shows a summary table comparing the best results of the four approaches. As for the low-level activities, one observes that the approach *Histograms + SVM* achieves the highest accuracy, in this case 91.8%. Combining the histogram features with an SVM instead of a nearest neighbor classifier leads to higher precision and recall values for all activities. Generally, the accuracy of all four approaches is over 80%, which is significantly higher than the chance level of about 33%. Even though the results might not generalize due to the small number of high-level activities in our set, we find that the high recognition rates are remarkable, considering the use of simple and easy-to-compute features in combination with a relatively large and challenging dataset.

6.6 Conclusion

The main goal of this chapter was to investigate how well current approaches in activity recognition can be applied to the recognition of high-level activities, which happen on the order of hours rather than minutes and consist of a diverse set of small scale activities. To this end, we recorded a naturalistic dataset with a user wearing three sensors on wrist, hip and thigh performing several instances of three different high-level scenes. We evaluated four different algorithms with respect to their ability to recognize both the low- and high-level activities contained in the dataset. One important aim of this research was to investigate to which extent current approaches for recognition of low-level activities can be directly applied to the recognition of high-level activities – i.e. using the same simple features without adding any intermediate levels of representation. We believe that

in the future such an approach would allow for scalable and efficient activity recognition systems based on simple sensors.

The results indicate that our algorithms can achieve competitive recognition rates for many of the low-level activities. The best results of slightly below 80% were achieved when using histograms of cluster assignments as features, combined with a support vector machine for classification. We investigated different window lengths and numbers of clusters and found that mapping the data to 50 clusters already leads to good results. In terms of sensor placement, using only two sensors at wrist and thigh resulted in equal or even better rates than using all three sensors.

When classifiying high-level activites, we achieve a recognition accuracy of up to 92%, which is clearly above the chance level of about 33%. We achieve these results with the same algorithms that we used for the low-level activities, merely by changing parameters such as the feature length and classification window. The best results were again obtained by the histogram-based approach in combination with an SVM. For all our approaches we use simple mean and variance features derived from accelerometer readings at 2 Hz. Considering the relatively simple sensors and features, as well as the challenging dataset, we find that the results for the high-level activities are surprisingly good.

We conclude that recognizing activities on such scales using only small and unobtrusive body-worn accelerometers is a viable path worth pursuing. Yet we are aware that the work presented in this chapter is but a first step towards recognition of high-level activities, and that more sophisticated models might yield better results. An obvious extension would be a hierarchical approach, using the outcome of the low-level classification as basis for the high-level inference. The next chapter explores a possible step in this direction, namely by using topic models to discover high-level structure in low-level activities.

7

Discovery of Daily Routines

In the last chapter we have seen that it is feasible in principle to use information from wearable sensors to recognize high-level structure in human activities. In this chapter we continue our work in this direction, by introducing a novel approach for modeling and discovering daily routines of a user from on-body sensor data. Inspired by machine learning methods from the text processing community, we convert a stream of sensor data into a series of documents consisting of sets of discrete labels. We then search for common *topics* or *activity patterns* in this data, using Latent Dirichlet Allocation. We show on real-world data that the discovered activity patterns correspond to high-level behavior of the user, are highly correlated with daily routines such as *commuting*, *office work* or *dinner routine*, and can be learned without any user annotation.

7.1 Introduction

Activity recognition has experienced increased attention over the years due to its importance to context aware computing in general and to its usefulness for application domains ranging from medical diagnosis over elderly care to human behavior modeling. This has resulted in various successful approaches that are capable to recognize activities such as walking, biking, sitting, eating or vacuuming. The majority of research has focused on activities that we termed *low-level activities* in the last chapter and that may be described and thus recognized by their respective body movements (such as walking and biking), body posture (such as sitting and eating), or object use (such as vacuuming). For many applications, however, the recognition of such simple activities is not enough. For instance in the case of elderly care it is interesting to recognize daily routines such as shopping or hygiene or in the case of office workers it is interesting to recognize routines such as attending a meeting, having lunch or commuting. What makes the recognition of such routines more complex is that they are typically composed of several activities and that the composition of activities has a large variability depending on factors such as time, location and individual.

This work introduces a novel approach to model and recognize daily routines such as commuting or office work from wearable sensors. For this we propose to leverage

the power of probabilistic topic models 1) to automatically extract activity patterns from sensor data and 2) to enable the recognition of daily routines as a composition of such activity patterns. This chapter shows that the novel approach can be applied both to annotated activity data as well as to the sensor data directly in an unsupervised fashion. When applied to annotated activity data, the automatically extracted activity pattern often correspond to daily routines. When applied to sensor data, the activity patterns allow recognition of daily routines with high accuracy while requiring only minimal user annotation. Therefore we argue that our approach is well suited both to minimize the amount of user annotation and to enable scalability to long-term recordings of activities.

The main contributions of this chapter are threefold. First, we propose a new method to recognize daily routines as a probabilistic combination of activity patterns. Second, we show that the use of probabilistic topic models enables the automatic discovery of the underlying activity patterns. And third, we report experimental results that show the applicability and the power of the approach to model and recognize daily routines even without user annotation.

The chapter is structured as follows. Next, we will motivate our approach and demonstrate its potential on a set of activity labels covering seven days of unscripted and real-world activity data. We will see that on the ideal set of ground truth labels, our method can model and identify activity patterns that correspond to high-level structure in the person's daily life. In Sections 7.2 and 7.3 we describe the technical details of our approach, and introduce the dataset that we used for evaluation. After that, in we introduce two different methods for extracting activity patterns from previously unseen sensor data: The first method (Section 7.4) uses supervised learning to assign activity labels to the sensor data. These labels are then used to identify activity patterns in an unsupervised fashion. The second method (Section 7.5) is completely unsupervised and uses clustering to generate a vocabulary of labels, which are then used for pattern extraction. We conclude with a summary and outlook.

7.2 Daily Routine Modeling using Topic Models

The activities we perform in our daily lives can be segmented and characterized on different levels of granularity. Which level to choose depends on the concrete application at hand, but there is evidence that we humans tend to structure and name these levels in a hierarchical fashion, and that at the lower and more fine-grained levels the structure is aligned with physical properties of the activities, such as motion and posture [Zacks and Tversky 2001]. Research in activity recognition exploits this fact by automatically naming the user's activity, based on low-level sensor data such as the acceleration of different parts of the body.

For many types of activities it is already sufficient to observe a small window of sensor data – usually in the order of seconds – to classify them with high confidence. The upper part of Fig. 7.1 shows a sequence of such activities as they were performed by a

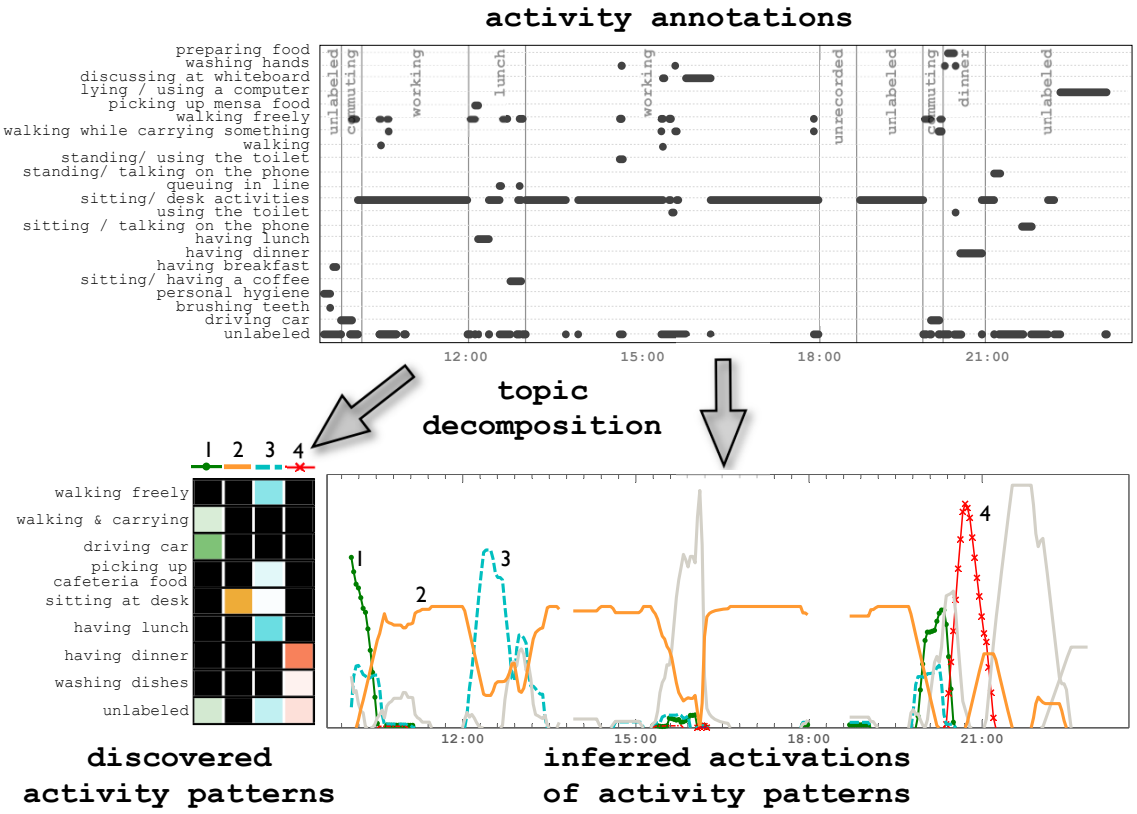


***Figure 7.1**: Top: Illustration of our approach on ground truth labels of activities. Note that the vertical high-level annotations (commuting, working, etc.) were not given to the algorithm. Lower Left: The matrix shows the contents of four out of ten discovered activity patterns. Lower Right: Inferred activations of the discovered activity patterns during the course of the day (e.g, the pattern in the third column is active during lunch time). Note the high correlation between these activations and the user annotated daily routines in the upper part, suggesting that these activations can be used to model daily routines.*

subject over the course of one day. If we were to further structure the activities the subject performed on this day, a natural approach would be to group them into *routines* such as *commuting*, *office work*, *lunch routine* or *dinner routine*. Such routines however cannot be identified from their local physical structure alone. What makes their recognition more complex is that they 1) are composed of variable patterns of multiple activities 2) range over longer periods of time, and 3) often vary significantly between instances. A model for recognizing such routines should be able to capture such facts as that *office work* "mostly consists of sitting", but "may (or may not) contain small amounts of using the toilet, or discussions at the whiteboard"; or that *commuting* "mostly consists of driving car, but usually contains short walking instances as well".

It turns out that a family of probabilistic models, commonly referred to as *topic models*, is well suited for this kind of task. Before giving more details about how to use and

infer topic models we first give an intuitive example what topic models can achieve when applied to activity data.

The lower part of Fig. 7.1 illustrates the result of our approach when applied to seven days of ground truth activity labels, including the sequence shown in the upper part of the figure. The columns of the matrix on the lower left represent four out of 10 different activity patterns or *topics* that were automatically identified by the method. Intuitively, each activity has a probability of occurring in the pattern, indicated by the color of the matrix cell. E.g., the third pattern (blue) is most likely to contain the activities *walking freely* and *having lunch*, and also - but slightly less - likely to contain *picking up cafeteria food*, *sitting at desk* and the *unlabeled* class. Similarly, the fourth pattern (red) is likely to contain *having dinner*, *washing dishes* and the *unlabeled* class.

For each of these activity patterns the method is able to tell how much each pattern is *activated* at each point in time. This is shown in the plot on the lower right, in which we plotted the activations of each of the 10 topics for the day shown in the upper part. One can observe that the third pattern (blue) is most active around lunchtime, and that the fourth (red) is active around dinner time. What makes this result remarkable is that no supervision was needed to infer both the activity patterns and their activations over the course of the day. The topic model essentially discovered these activity patterns in a entirely unsupervised way. In this particular case the activations of these activity patterns are highly correlated with the daily routines of the person.

While in this particular example the activity patterns and the daily routines have been discovered in an unsupervised way, they required as input the activity annotations from the user. While this shows the principle applicability of topic models to model daily routines, it is clearly desirable to avoid the time-consuming and error-prone task of manual annotation. Later in the experimental sections we show that topic models can be applied to activity recognition results as well as to sensor data directly. In the latter case no user annotation is required whatsoever and still the discovered daily routines have a high correlation with the user-annotated daily routines.

7.2.1 Topic Models

Topic models stem from the text processing community [Hofmann 2001, Blei *et al.* 2003]. They regard a document - e.g. a scientific paper - as a collection of words, discarding all positional information. This is called a “bag-of-words"-representation. As single words capture a substantial amount of information on their own, this simplification has shown to produce good results in applications such as text classification. Assume, for example, an author wants to write a UbiComp paper that covers the three topics “HCI”, “Elderly Care” and “Context-Aware Computing”. Writing this paper then is essentially picking N times a topic from his list of chosen topics and then picking a word appropriate for the topic. Therefore he uses a probability distribution that tells which words are likely to be used for which topic. Different topics might share certain words, which means that both topics

assign a high probability to them. This process yields a document in the "bag-of-words" representation with N words.

Most interestingly for the purpose of the work presented in this chapter, topic models allow to infer the inherent topics from an appropriate corpus of documents. E.g. when applied to the corpus of all UbiComp papers one expects that topics such as “HCI”, “Elderly Care”, and “Context-Aware Computing” (among many other topics) can be discovered automatically without any user annotation or intervention. To illustrate how this can be achieved, we describe the process of writing the documents in a bit more formal way. As mentioned before, the author of document d picks a set of topics. Assuming that he puts different emphasis on the different topics, we model the mixture of topics as (multinomial) probability distribution $p(z|d)$ over topics z. Similarly, the importance of each word for each topic z is also modeled as a (multinomial) probability distribution $p(w|z)$ over words w of a vocabulary. Given these two distributions, we can compute the probability of a word w occurring in document d:

$$p(w|d) = \sum_{z=1}^{T} p(w|z)p(z|d), \tag{7.1}$$

assuming that there are T topics the documents - e.g. all UbiComp papers - are dealing with. This probability distribution $p(w|d)$ doesn't include any notion of topics any more and in fact can be estimated by simple counting of the words in each document. Having many documents, we observe a data matrix of observed $p(w|d)$ as depicted on the left hand side of the equation in Fig. 7.2. According to Equation 7.1 (which is equivalent to the described process of writing the paper), the data matrix can be reconstructed by a matrix product of the word relevances for each topic and a mixture of topics $p(z|d)$ for each document. Estimating the topic model means doing the reverse. The data matrix on the left-hand side is decomposed into the two matrices on the right-hand side, thereby recovering the characteristic words for each topic and the mixture of topics for each document.

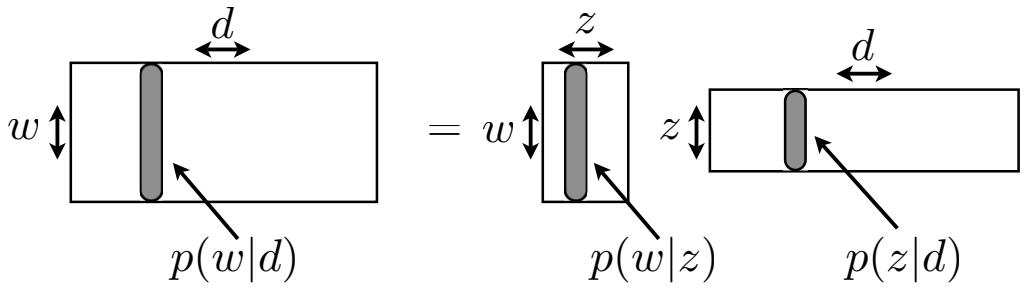


***Figure 7.2**: Intuition of topic model decomposition. By introducing an unobserved, latent topic variable z, the observed data matrix of $p(w|d)$ is decomposed into a topic-word matrix of $p(w|z)$ and a document-topic matrix of $p(z|d)$*

The described formulation addresses precisely the task we formulated earlier. The data matrix $p(w|d)$ corresponds to the activity data depicted in the upper half of Fig. 7.1 and the decomposition illustrated in Fig. 7.2 corresponds to the activity patterns $p(w|z)$ and activations of activity patterns $p(z|d)$ in the lower half. Therefore we propose to discover activity patterns as topic-word distribution and daily routines by topic activation.

In the following experiments we use a particular instantiation of these kind of models - called Latent Dirichlet Allocation (LDA) [Blei *et al.* 2003], that extends the described pLSA model to a Bayesian approach by placing a dirichlet prior $p(\theta_d|\alpha)$ with parameter α on the document-topic distributions $p(z|\theta_d)$. Fitting the model is equivalent to finding parameters α for the dirichlet distribution and parameters β for the topic-word distributions $p(w|z,\beta)$ that maximize the likelihood $\mathcal{L}$ of the data for documents $d = 1,\ldots,M$:

$$\mathcal{L}(\alpha,\beta) = \prod_{d=1}^{M} \underbrace{\int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \underbrace{\sum_{z=1}^{T} p(w_n^d|z,\beta)p(z|\theta_d)}_{\text{marginalize over } z} \right) d\theta_d}_{\text{marginalize over topic activations } \theta_d},$$

where T is the number of topics and each document d consists of the words w_n^d with $n = 1,\ldots,N_d$. For a more detailed description of the learning process the reader is referred to the original paper by Blei et al. [Blei *et al.* 2003]. We also use their implementation, available at [Blei 2006].

7.3 Dataset

To show the effectiveness of the approach, we recorded the daily life of one person over a period of sixteen days. The subject was provided with two wearable sensors, one of which he placed in his right hip pocket, the other on the dominant (right) wrist. The recordings were started in the morning shortly after getting up, and usually ended in the late evening before going to bed. This enabled us to record continuous, non-scripted activities in a natural environment. Due to memory constraints of the sensor platform, the memory had to be emptied after about 4 hrs of recording. A recording of one day typically consists of three such parts, i.e. roughly 12 hrs of data with two gaps in between. In total, our dataset consists of 164 hrs of recordings. Of these, we had to discard 28 hrs due to failures in the sensor hardware. Figure 7.3 gives an overview of the recordings.

Sensor Hardware. Fig. 7.4(a) shows the Porcupine sensor platform [Van Laerhoven *et al.* 2006] which we used to record our set of activities. Besides a 3D accelerometer (ADXL330) and a PIC microcontroller which we used for preprocessing of features, it includes a realtime clock, nine binary tilt switches, a temperature sensor and two light

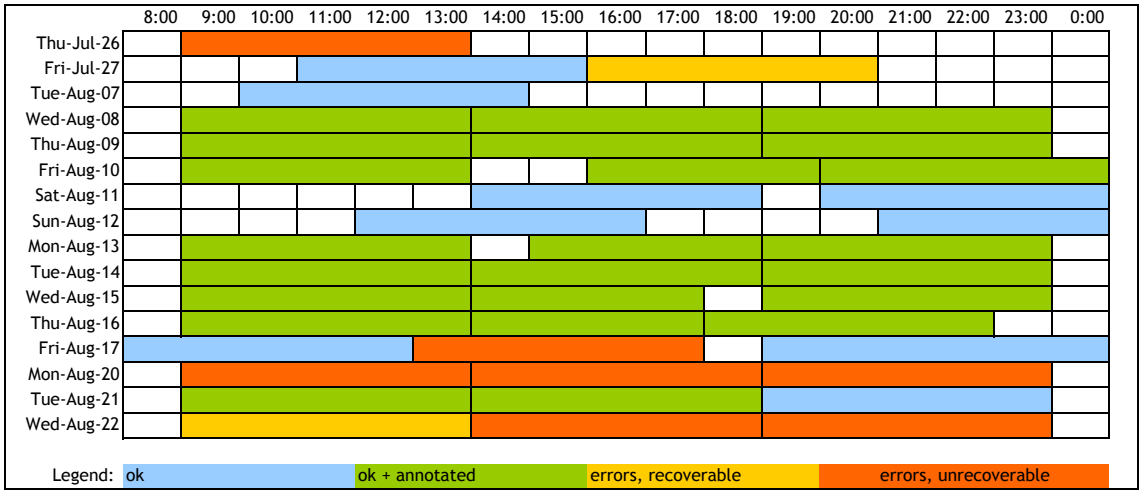
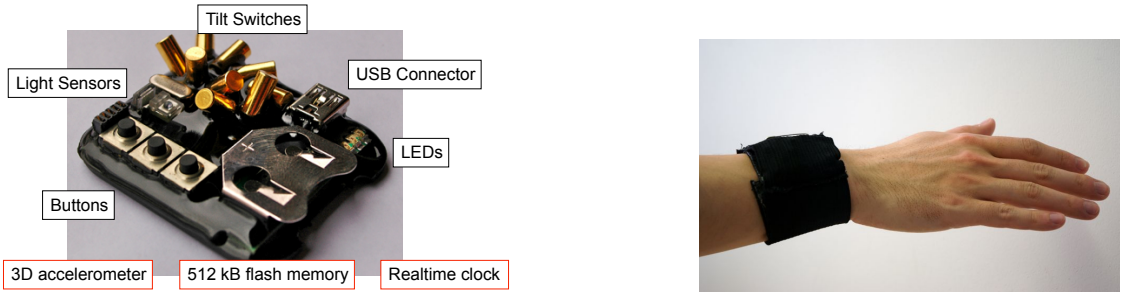


***Figure 7.3**: Overview of the recorded dataset. Two factors made the recording process difficult, namely the relatively small size of the onboard memory of the sensors, and occasional sensor failures. A single recording consists of about four hours of data, after which the contents of the memory had to be emptied. This resulted in occasional gaps in the coverage that can be seen in the figure.*



(a) (b)

***Figure 7.4**: The wearable sensor platform used for recording activities.*

sensors. Data can be stored on 512 kb of flash storage and transferred via a USB connector. In addition, the device features three buttons and three LEDs which can be freely programmed, e.g. for annotation or status display. The platform is small and light enough to be comfortably worn on a wristband (Fig. 7.4(b)) or slid into the subject's pocket.

Features. The sensors deliver data at a rate of roughly 100Hz. Due to the memory constraints we subsampled the data by calculating mean and variance over a sliding window of 0.4 seconds (i.e. @ 2.5Hz), and store them along with a timestamp from the realtime clock. This allows to store about four hours of sensor data on the onboard memory of the sensor.

Annotation. To analyze the effectiveness of our approach, we aimed for two different levels of annotations. First, we asked the user to annotate daily routines such as *commuting* or *working*. And second, we also aimed to obtain detailed annotations of the individual activities – at least for part of the data. In total we annotated seven complete days (84 hrs) in detail, which we used for our experiments. In the experiments reported below we will analyze the recognition of daily routines both with and without these detailed annotations of individual activities. This allows us to show that the approach is not only applicable in supervised settings but also in entirely unsupervised settings.

Finding a good balance between detailed and complete annotations and minimal user disruption is a common problem of studies in activity recognition, especially for long-term studies outside a laboratory. We used a combination of several online and offline annotation methods so that the user had some freedom to choose a method that suited him depending on the situation. Online annotation takes place while the activities are being recorded. We employed three different methods of online annotation, namely experience sampling, a time diary and camera snapshots. During experience sampling, the subject was notified in periodic intervals by an application running on his mobile phone, which presented a set of questions about his current activities. The time diary is a handwritten log in which the subject entered the names, start- and ending times of activities. As a third method, the subject took occasional snapshots with the built-in camera of the mobile phone.

It turned out that for our setting the time diary was the most useful online annotation method, providing detailed information while being far less disrupting than expected. One likely reason for this is that the subject was often working near or at a laptop, which he could use to quickly log activities. Our experience sampling application, while relatively fast and easy to use, tended to miss short events, pose redundant queries and was less precise than the time diary in determining start and ending times of activities. For offline annotation, we visualized the sensor data and aligned it with the annotations from the experience sampling application and the time diary, as well as with the photographs taken by the subject, and had the subject fill in remaining gaps, refine start- and ending times of activities, and also identify and annotate daily routines.

Recorded Activities. Our subject annotated a total of 75 distinct activities and daily routines. For our evaluation, we filtered out activities that occurred only once or for very short durations, and merged similar ones into single classes. Within the individual activities and within the daily routines there is no overlap between annotations, and for both sets we introduced an additional *unlabeled* class, so that in the end each feature is assigned to one activity and one daily routine.

The activity set consists of the following 34 activities, along with the *unlabeled* class (duration in minutes shown in brackets): *sitting / desk activities (3016.9), lying while reading / using computer (196.6), having dinner (125.3), walking freely (123.6), driving car (120.3), having lunch (75.2), discussing at whiteboard (62.6), attending a presentation (48.8), driving bike (46.2), watching a movie (42.5), standing / talking on phone*

Online			*Offline*
Experience Sampling	*Time Diary*	*Camera Snapshots*	*Indirect Observation*
User answers periodic queries (every 15-30 min) about his current activity, on mobile phone	User writes down current activity along with timestamp, either electronically or on paper	User takes snapshots with mobile-phone camera	Sensor data & online annotations are presented to the user; user refines / adds annotations
☺ user is reminded to annotate (less awareness required, better coverage)	☺ no redundancy or interruptions	☺ fast & easy	☺ can lead to very detailed annotations
☺ fast & easy to use	☺ user can decide on necessary level of detail	☺ good as rough reminder	☹ errors may be introduced
☹ unknown events not covered	☺ can handle unknown / complex activities	☹ some activities not obvious from pictures	☹ can be time-consuming depending on level of detail
☹ short events may be missed	☹ requires increased awareness	☹ start / end of activity not covered	
☹ redundant queries can be annoying	☹ feasibilty depends on type of activities; impossible in some situations	☹ may be inappropriate in some situations	
☹ may be interrupting/ embarrassing depending on the situation	☹ user may forget to annotate		

Figure 7.5*: We used several online- and offline annotation methods during data collection. The table summarizes our positive and negative experiences with each of the methods. We found that a combination of several methods leaves the user the choice to decide on the most appropriate method, depending on the situation, and can also lead to a more complete and detailed coverage. In general, the most appropriate method will depend on the type of scenario.*

(24.8), walking while carrying something (22.8), walking (22.8), picking up cafeteria food (22.6), sitting / having a coffee (21.8), queuing in line (19.8), personal hygiene (17.2), using the toilet (16.7), fanning barbecue (15.2), washing dishes (12.8), kneeling / doing sth. else (11.6), sitting / talking on phone (8.7), kneeling / making fire for barbecue (8.2), setting the table (8.0), standing / having a coffee (6.7), preparing food (4.6), having breakfast (4.6), brushing teeth (4.3), standing / using the toilet (3.0), standing / talking (2.8), washing hands (2.1), making coffee (1.8), running (1.0), and wiping the whiteboard (0.8)

Four daily routines (plus the *unlabeled* class) have been annotated that span longer periods of time, typically dozens of minutes to several hours, and which are composed of several distinct activities. The first routine is *commuting (289 min)*, which includes leaving the house and driving to work either by car or by bike, until arriving at the office, and vice versa in the evening. The longest routine is *office work (2814.7 min)*, which mainly comprises desk activities, with occasional interruptions, e.g. when fetching a coffee, visiting an office mate, going to the toilet, attending a meeting, etc. At noon the subject usually went to a nearby cafeteria to have lunch, followed by a stop at a neighboring coffee place. This episode, which usually lasted about an hour per day, is labeled as *lunch routine (391.3 min)*. The last routine is *dinner activities (217.5 min)*, which mostly includes setting the table, having dinner and washing the dishes. As all

of the recorded days are weekdays, these four daily routines cover a large percentage of the data, leaving out only some parts in the mornings and evenings. Figure 7.6 shows an example of one day of sensor data, along with some snapshots taken by the user's mobile phone.

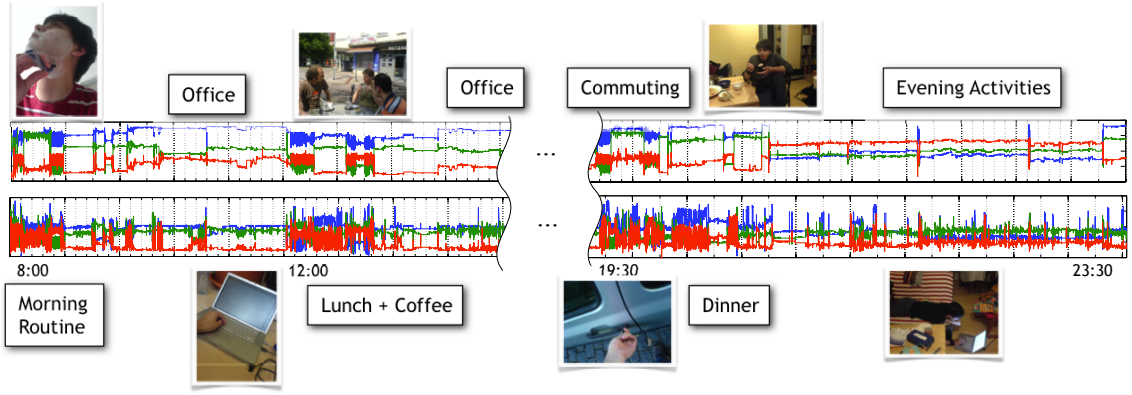


***Figure 7.6**: One day of sensor data as used for our experiments. The plots show the mean of the acceleration in x-, y- and z-direction at the wrist (lower row) and pocket (upper row).*

7.4 Discovery of Daily Routines based on Activity Recognition

As discussed earlier, the proposed approach using topic models can be used to model daily routines based on wearable sensors. While the example in section 7.2 has relied on user generated annotations to discover activity patterns this section uses supervised activity learning to generate and recognize a vocabulary of activities. Based on the recognized activities topic models are then used to first learn and discover activity pattern which are then in turn used to describe and recognize daily routines. Therefore we first describe how we train a supervised classifier on labeled data, and then use the labels obtained from the classifier as vocabulary for topic estimation. The next section will then describe how the vocabulary can be obtained in an unsupervised way thereby making the approach scalable to large amounts of training data.

Activity Recognition. We evaluated several combinations of features and classifiers on our set of activities. From the acceleration signal we computed several features, including mean, variance, and a number of frequency features, over sliding windows between 0.4 and 4 seconds. As additional feature we used the time of day provided by the realtime clock of the sensor. As classifiers we evaluated SVMs, HMMs and Naïve Bayes. They are standard representatives of discriminative and generative classifiers and have been

successfully used before for similar tasks (e.g. [Oliver *et al.* 2002, Huỳnh *et al.* 2007]). All our results are cross-validated in a leave-one-day-out fashion.

We first compared the three classifiers and different features on a subset of the data spanning two days. It turned out that due to the size of the dataset, SVM and HMM training and classification took significantly longer than Naïve Bayes. Due to its time efficiency and since the overall recognition accuracy of Naïve Bayes was only marginally lower, we settled for this approach, using as features mean and variance of the 3D-acceleration signal from wrist and pocket motion, plus the time-of-day information from the realtime clock (adding up to a 13-dimensional feature vector). The use of frequency features did not improve the results in our setting, which may be due to the relatively coarse resolution (2.5Hz) of the data.

Overall we achieved an accuracy of 72.7% on the activity dataset. The individual results vary considerably, owing to the diversity of the collected activities. The best five results were obtained for *sitting/ desk activities* (precision 89.5%/ recall 95.4%), *walking freely* (96.2/ 84.2), *standing/ talking on phone* (82.8/ 96.4), *driving bike* (96.7/ 77.1), *having lunch* (76.5/ 97.5) and *personal hygiene* (89.0/ 66.2). A number of activities only occurred on one day, so that the classifiers had no chance of classifying them correctly in our leave-one-day-out crossvalidation protocol. Among these were *kneeling, running, standing while having a coffee, wiping the whiteboard*, and *attending a presentation*. Most of the activities with low recognition scores were either very short (e.g. *washing hands* (precision 3.7%/ recall 3.4%)), so that only little training data was available, or they were confused with other, similar activities (eg. *sitting/ having a coffee* (22.4/ 35.6) was often confused with *desk activities*),

The time-of-day feature enables the classifier to separate activities which share a common motion signature but are performed at different parts of the day – the main improvement in recognition could be observed for the two activities *having lunch* and *having dinner*, for which confusion was virtually eliminated and precision/recall scores improved from 16.8%/ 20.5% to 38.9%/ 47.9% (*having dinner*) and from 62%/ 19% to 97.5%/ 76.5% (*having lunch*). As we used unimodal gaussians to model the Naïve Bayes likelihoods, the time-of-day feature did not worsen the results for activities which occur at irregular times during the day – if enough data for such activities exists, then the larger variance usually flattens the likelihood function to a point at which it has little influence on the final posterior. Problems with the time-of-day feature may arise for activities which are not time-dependent but occur only few times in the training data.

7.4.1 Topic Estimation based on Activity Recognition

As a result of our supervised training procedure, we obtain for each data sample a posterior probability for each activity, along with a discrete label that corresponds to the activity with the highest posterior. Our next goal is to discover daily routines from this stream of low-level data, using the framework provided by Latent Dirichlet Allocation.

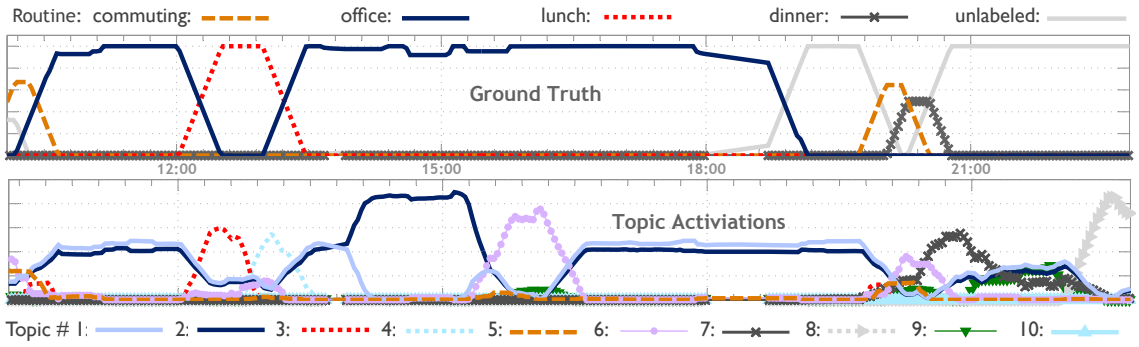
The main choices one has to make when applying LDA to activity data are the nature and size of the vocabulary, the size of the documents and the number of topics. A simple yet effective way to create documents for the topic models from a stream of activity labels is to use a sliding window of length D over the labels and construct for each window a histogram of label occurrences. In this way each document represents a mixture of activities over a window of time. We found that the outcome of the topic estimation process can be made more robust to noise and misclassifications of the underlying classifier by generating the vocabulary not from the hard assignments of the classifier, but from the soft assignments given by the posterior probabilities for each activity. We achieve this by summing up the posterior probabilities for each activity over the size of one document window, and then generating labels for each activity in proportion to the sum of all posteriors.

Qualitative Results. Figures 7.7(b) and 7.7(a) (bottom) show the result of LDA estimation and inference when generating documents over windows of 30 minutes, shifted by 2.5 min at a time. In this example we chose $T = 10$ topics and set the dirichlet prior α to 0.01. The topics in Fig. 7.7(b) were estimated from six days of data. For each topic z we list all activity labels w with $p(w|z) \geq 0.01$. Fig. 7.7(a) (bottom) shows the activations of those topics on the day that was left out during training. In each time step we plot the topic activations that correspond to the document covering the preceding 30 minutes.

The first important observation which can be made from the results shown in Fig. 7.7 is that there are topics that clearly correlate with the daily routines of the subject's day. This can be seen by comparing the topic activations to the daily routines annotated by the subject (Fig. 7.7(a)). To see how well the estimated topic activations correspond to the mixture of ground truth labels in the respective time window, we also collected the ground truth labels in sliding windows the same size as the documents, i.e. we assigned to each time step the percentage that an activity was 'active' during the last 30 min.

Topics 1 and 2 are both active during office work so that their joint or individual activation is a good indication of office work. In the afternoon topic 6 is activated strongly for a certain period of time, corresponding - on that particular day - to a presentation of a colleague. Topic 6 is a good example of a newly 'discovered' routine – it does not appear in the annotations of the user's daily routines, yet it represents a valid activity pattern that can be modeled and identified. The lunch routine is represented by two topics, namely 3 and 4. As the typical lunch routine is composed of a visit to the cafeteria and the visit of a cafe, topics 3 and 4 have captured the differences in these two "phases" of the lunch routine. Again the activation of either of these topics allows the recognition of the lunch routine. The dinner routine is correlated with the activation of topic 7. The remaining daily routine, *commuting*, is not directly correlated with a single topic but rather with a combination of topics. Both in the evening and in the morning the co-activation of various topics including topics 5, 6 and 3 allow to identify this routine.

Let's now turn to the contents of the topics, i.e. the learned activity labels that have a high probability of being part of a particular topic. As can be seen from Fig. 7.7(b), the



(a)

1 sitting / desk activities (1.0)
2 sitting / desk activities (0.99)
3 having lunch (0.5), walking freely (0.26), picking up cafeteria food (0.06), queuing in line (0.06), unlabeled (0.04), walking & carrying sth (0.02), brushing teeth (0.02)
4 standing / using toilet (0.69), walking freely (0.18), queuing in line (0.06), walking (0.02), desk activities (0.02)
5 driving car (0.33), walking & carrying sth. (0.21), sitting / desk activities (0.2), walking (0.14), unlabeled (0.08)
6 using the toilet (0.69), walking freely (0.17), discussing at whiteboard (0.06), sitting / desk activities (0.03), brushing teeth (0.02)
7 having dinner (0.77), desk activities (0.1), washing dishes (0.08), unlabeled (0.04)
8 lying / using computer (1.0)
9 unlabeled (0.87), driving bike (0.04), washing dishes (0.02), stand/ use toilet (0.02), washing hands (0.02), standing / using the phone (0.02)
10 watching a movie (1.0)

(b)

Figure 7.7*: (a) Ground truth and topic activations for one day, based on a vocabulary of learned activity labels. (b) Contents of the ten estimated topics. The numbers in brackets indicate $p(w|z)$, i.e. the probability of the activity label w given the current topic z (labels w with $p(w|z) < 0.02$ are not shown). The distributions were estimated from six days of data. (a) shows the inferred topic activations for the day that was left out during training.*

content often represents a meaningful set of activity labels. E.g., the prominent words in topic 3 are having lunch, walking, picking up cafeteria food and queuing in line. Topic 5 is a mixture of driving car, walking, and desk activities and is activated during the *commuting* routine of the subject. Topics 1 and 2 represent desk activities and are active during the *office work* part of the subject's day. Topic 7 contains having dinner and washing dishes, as well as desk activities and driving car, which all correspond to evening and dinner activities of the subject.

Since the accuracy of the underlying classifier that generates the vocabulary is not perfect, there are errors due to misclassifications, some of which are reflected in the contents of topics. E.g., the classifiers for the activities *using the toilet* and *standing / using the toilet* fire relatively often, but only with precision of 18% and 7%, respectively. Partly due to a small amount of training data, they are often confused with similar activities such as *desk activitites* and *standing at the whiteboard*. As a consequence, in the example shown in Fig. 7.7, their labels are weighted too strong in topics 4 and 6. A more powerful activity recognition algorithm would help to alleviate such problems even though it is expected that significant ambiguities between activities remain. An important and relevant property of topic models is that they are robust to these types of ambiguities.

Evaluation Method. While the plots of the topic activations suggest that the topics are indeed able to discover and model activity patterns and therefore high-level structure in the subject's daily activities, it is not obvious how to quantify the results. We propose two different measures for evaluating the quality of the topic decompositions: correlation and recognition performance. For both measures we use as ground truth the daily routine annotations by the subject. First it should be noted, though, that both methods are not optimal, since LDA is an inherently unsupervised method which is able to discover meaningful structure a user was previously unaware of. Such ability cannot be quantified when evaluating against a predetermined ground truth.

For the correlation measure, we first perform LDA estimation on six of the seven recorded days. We then assign to each activity the topic to which the correlation to the ground truth annotation is highest. Next we perform LDA inference on the seventh day and note for each activity the correlation with its assigned topic. We repeat this in a leave-one-day-out fashion and report the average results for each daily routine. In order to compute recognition performance, we use the topic activation vectors as features for a supervised learning task. More specifically, we first perform LDA estimation and inference on six of the seven days, and then train a nearest neighbor classifier using the obtained topic activation vectors and the daily routine ground truth. We then perform LDA inference on the seventh day and classify each of the resulting activation vectors using nearest neighbor. The results we report are again cross-validated over the seven days of data.

Baseline Results. In order to obtain a baseline for the recognition of routines, we built a supervised classifier using HMMs based on the same features that we use for the LDA-approach, i.e. acceleration features from wrist and pocket sensor, plus time-of-day. We used left-right models and varied the number of states Q, the number of gaussians per state M, as well as the length of the observation sequence O. The cross-validated results for the best parameter-combination that we found ($Q = 5$, $M = 2$, $O = 30min$, shifted by $5min$) are shown in Fig. 7.8. The *lunch* and *office work* routines can be predicted with high precision and recall. *Lunch* is a short, yet very regular routine, usually taking place between noon and 1pm. *Office work* covers a large part of the day and consists to a large part of sitting activities. In contrast, *dinner* and *commuting* are relatively short routines that occur at relatively irregular times of day, which makes recognition more challenging. This is reflected in the lower recall values. In the remainder of the paper we will use these results as a baseline for the recognition of routines using topic models.

Quantitative Results. Fig. 7.9 shows the correlation and recognition results for the best combination of parameters when we used learned activity labels as vocabulary. In this case we used $T = 10$ topics, a document length of 30 min, and soft assignments from class posteriors to generate the words for each document. *Office work* is best correlated and recognized, followed by *lunch*, *commuting* and *dinner*. Comparing to our baseline results (Fig. 7.8), we can see that the recognition of routines has improved. The values for precision and recall increase throughout, with the exception of precision for *dinner*

Routine	*Precision*	*Recall*
Dinner	88.6	27.3
Commuting	72.6	31.5
Lunch	84.4	80.7
Office Work	89.2	91.1
Mean	83.7	57.7

***Figure 7.8**: Baseline recognition results, using HMMs based on acceleration and time-of-day features.*

routine. Overall, the results indicate that the estimated topics relate to high-level structure in the subject's daily routine.

Influence of Parameters. For the daily routines in our data set, correlation with topics dropped noticeably when choosing document windows smaller than 30min. In general our results indicate that choosing document lengths on the order of the average lengths of the routines seems a good strategy. We also found that using more topics may lead to better recognition results when using topics activation vectors as features, but makes (visual) discovery of unknown routines more difficult, as the topic activation plots get more noisy.

Routine	*Correlation*	*Precision*	*Recall*
Dinner	0.7	75.5	40.2
Commuting	0.6	85.5	51.8
Lunch	0.8	87.0	83.3
Office Work	0.8	96.4	93.7
Mean	0.7	86.1	67.2

***Figure 7.9**: Correlation and recognition results when using topics estimated from learned activity labels.*

7.5 Unsupervised Learning of Daily Routines

In the previous section we showed how topics can be used as a means of inferring high-level structure from a vocabulary of labels representing relatively short-term activities. These labels were learned in a supervised fashion from a stream of sensor data. An advantage of this approach is that the estimated topics carry an inherent meaning, which is expressed by the distribution of labels within each topic. A substantial disadvantage, though, is the amount of annotation effort associated with the supervised learning part. In this section we describe how the vocabulary for the topic estimation can be constructed in an unsupervised fashion. We will show that surprisingly good results can be obtained without any need of tedious and detailed activity annotation.

Clustering of Activity Data. To generate discrete labels from continuous sensor data in an unsupervised fashion we simply use data clustering. This allows to assign to each sample the index of the closest cluster centroid. While this is essentially the basis of our approach, we again found that using soft instead of hard assignments did improve our results. In order to create a vocabulary of size N, we first cluster our feature vectors using K-means clustering with $K = N$. For each feature i we store the distances $d_{1..N}$ to the centroids of each cluster. We then convert these distances to weights $\omega_{1..N}$ with

$$\omega_i = \frac{e^{-\frac{d_i}{\sigma}}}{\sum_{j=1..N} e^{-\frac{d_j}{\sigma}}} \tag{7.2}$$

Thus smaller distances imply higher weights, and the weights for one feature sum up to one. The parameter σ controls how fast the weights decline for more distant clusters. Empirically we found that setting σ to the standard deviation of all distances worked well. We next use the weights to construct documents of size D in the same fashion as for the supervised case described in the previous section. More specifically, for each cluster i we sum up the weights ω_i over a feature window of length D, and then generate m_i labels for this cluster by multiplying the sum of its weights by the document length D and rounding to the next integer. Since the weights for each feature are a partition of 1, the document will contain at most $\sum m_i = D$ labels.

Results. Fig. 7.10 shows an example of the result of LDA inference using a vocabulary of 10 cluster labels, together with the daily routine ground truth for this day. The documents were created from sliding windows of 30 min, shifted by 2.5 min at a time. LDA estimation was performed on six of the seven days, and inference on the remaining day. Again one can observe that the topic activations reflect the annotated daily routine structure of the subject's day, even though this time no annotations (neither for activities nor for daily routines) were given at all. Furthermore, there are individual topics whose activation is strongly correlated with the *lunch*, *office work* and *commuting* routines.

Fig. 7.11 shows correlation and recognition scores for the best combination of parameters when using a vocabulary of cluster labels for topic estimation. In this case we used $T = 10$ topics, a document length of 30 min, and $N = 60$ clusters. Note that low correlation does not necessarily imply bad recognition performance, as can be seen for the *commuting* activity. This is because we compute correlation between individual topics and daily routines, while recognition uses the activations of all topics at each time step. Thus if a daily routine can be characterized by a mixture of topics instead of a single topic, recognition scores may be high even though the best correlation of an individual topic is low.

Comparing the results to the supervised method described in the last section (Fig. 7.9), one can observe that the mean correlation and precision are lower in the unsupervised case, with about 10% less overall precision and a drop of 0.1 in correlation score.

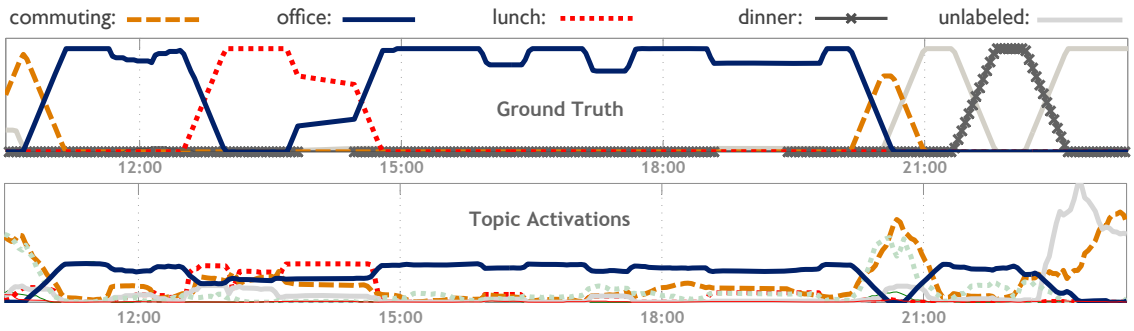


***Figure 7.10**: Top: Daily routine ground truth for one day of data. Bottom: Inferred topic activations, based on a vocabulary of ten cluster labels. Ten topics were estimated from six days of data, and the plot shows the activation of these topics on the day that was left out during training.*

However, overall recall declines only slightly, and the individual recognition scores for *office work* and *commuting* remain high. One likely reason for the drop in precision for the *lunch* and *dinner* routines is that they share many activities and are therefore not separated well by the clustering. As a consequence, recognition of *lunch* drops below our baseline results. However, compared to the baseline, recall for *dinner* and *commuting*, as well as precision for *commuting* are higher, indicating that the approach can compensate for the irregular occurrences of these routines. Finally, keep in mind that these results are based on predefined ground truth, and thus do not capture the ability of the method to discover previously unknown structure in the data.

Routine	*Correlation*	*Precision*	*Recall*
Dinner	0.6	56.9	40.2
Commuting	0.5	83.5	71.1
Lunch	0.8	73.8	70.2
Office Work	0.6	93.4	81.8
Mean	0.6	76.9	65.8

***Figure 7.11**: Correlation and recognition results when using topics estimated from k-means cluster labels.*

7.5.1 Discussion

In this section we used clustering as an unsupervised method to generate a vocabulary of discrete labels from a stream of continuous activity data. We used this vocabulary as basis for topic estimation and observed that the estimated topics correlate with daily routine structure in the subject's activities. The main advantage of this approach is that it does not require any labeled training data and yet is able to discover structures that are of relevance to the subject. As the approach is entirely data-driven, we don't rely on any noisy classifier output, and hence there are no 'wrong' words that the topic model has to

deal with, as we observed in the supervised case. On the other hand, the contents of the topics, i.e. the distribution over cluster labels, carries no direct meaning for an observer. Such meaning can be established, however, via the additional step of comparing the topic activations to the actual structure of the subject's day, and then identifying topics that correspond to possible daily routines.

Figure 7.12 gives a conceptual overview of the approach that we described in this section and the alternative approach described in Section 7.4, and compares them in terms of the amount of supervision required.

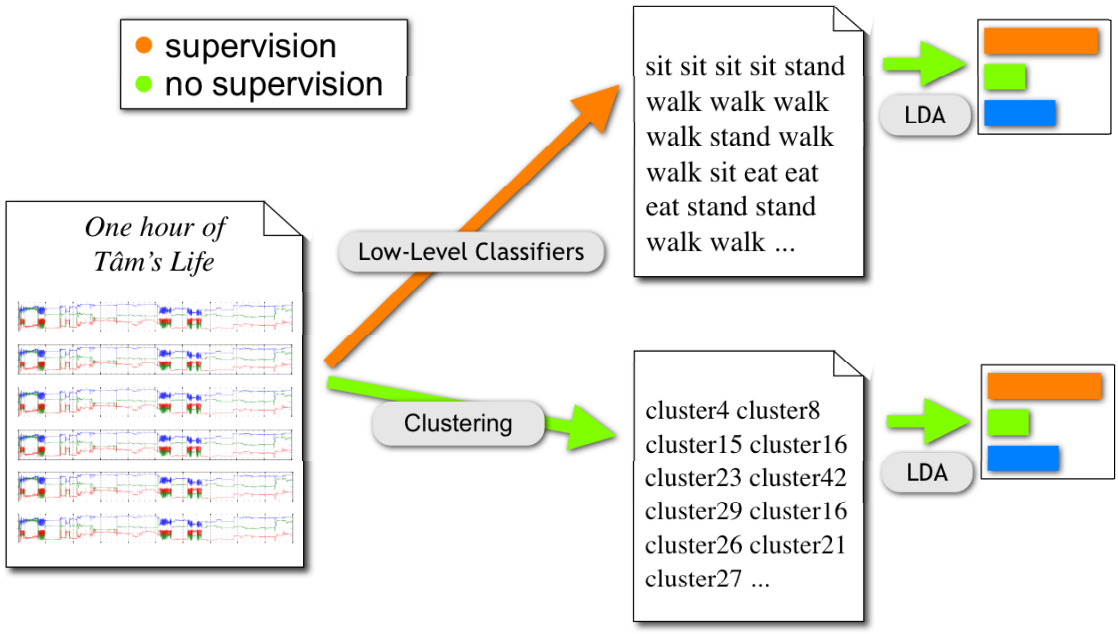


***Figure 7.12**: Overview of our two approaches to discovery of activity patterns from sensor data. Starting from a stretch of sensor data (left), we generate either a set of activity labels (upper path; Section 7.4) or a set of cluster assignments (lower path; Section 7.5). The sets correspond to the documents required by the subsequent topic estimation step.*

7.6 Conclusion

In this chapter we have introduced a novel approach for modeling and discovering daily routines from on-body sensor data. Inspired by machine learning methods from the text processing community, we convert a stream of sensor data into a series of documents consisting of sets of discrete activity labels. These sets are then mined for common topics, i.e. activity patterns, using Latent Dirichlet Allocation. In an evaluation using seven days of real-world activity data, we showed that the discovered activity patterns correspond to high-level behavior of the user and are highly correlated with daily routines such as *commuting*, *office work* or *dinner routine*.

The patterns can be based on a learned vocabulary of meaningful activity labels (such as *walking*, *using the phone*, *discussing at whiteboard*, etc.), in which case the discovered patterns are immediately human-readable in that they represent sets of such labels. Learning of labels requires a supervised component, which can be avoided by applying our method directly to unlabeled sensor data using clustering. In this case, the method is fully unsupervised, yet still allows to visualize high-level structure of the data, as well as to identify activity transitions, novelties and anomalies.

We think that both the (partly) supervised and the unsupervised approach have advantages and limitations, which should be considered in the light of specific application scenarios. Moreover, the approaches need not necessarily exclude each other. E.g., the unsupervised approach can help to detect anomalies, but not necessarily tell what exactly happened (e.g. *getting up at night to go to the toilet*, vs. *getting up to sleepwalk*). This could be addressed by the use of semi-supervision, e.g. by presenting the user with a visualization of topic activations such as in Fig. 7.10 (bottom), and asking him to label the discovered topics.

In conclusion, we believe that our approach is highly appealing for the field of activity recognition, and that so far we have only exploited some of its potential. E.g., as can be seen from the topic activation plots, the probabilistic nature of the approach allows for handling of concurrent and overlapping activities (expressed as co-activation of patterns), and also transitions between activities. We consider these properties, together with the ability to decompose routines into their low-level constituents, as a crucial advantage over traditional unsupervised techniques such as clustering.

8

Conclusion and Outlook

Context-aware computing is a broad field of research. This thesis has investigated one of its many facets, namely activity recognition with wearable sensors. In the following we summarize our main findings and contributions, and then give an outlook on possible future work.

Choosing appropriate features can improve recognition. In Chapter 3, we have presented a systematic comparison of commonly used features for activity recognition. Our results indicate that the choice of features can be crucial for the success of a recognition algorithm, and prior to our study there existed little work on evaluation of features with respect to recognition of specific activities.

We have developed new unsupervised and semi-supervised learning methods for activity recognition. Reducing the amount of supervision in activity recognition is important for developing scalable systems that can adapt to new users and scenarios with minimal annotation overhead. Towards this goal, we have introduced a novel approach for unsupervised learning of activities from low-level sensor data in Chapter 4. The proposed approach is neither limited to specific activities nor specific types of sensors. We described the algorithm, proposed an extension to multiple time scales, and evaluated the approach on several data sets, showing that it can be used to reliably model and also recognize activities. In Chapter 5 we have extended the approach to allow for semi-supervised learning, by proposing to combine it with a discriminative classifier. The generative part of the algorithm allows to extract and learn structure in activity data without any prior labeling or supervision. The discriminant part then uses a small but labeled subset of the training data to train a discriminant classifier. Experiments showed that this scheme enables to attain high recognition rates even though only a subset of the training data is used for training. In addition, we analyzed and discussed the tradeoff between labeling effort and recognition performance.

Modeling and recognizing high-level activities can be achieved with low-level sensors. This thesis has taken a first step towards modeling and recognizing high-level

activities from body-worn accelerometers. In order to find out in how far traditional methods for recognizing low-level and short-term activities can be scaled to the recognition of high-level activities, we conducted a study using 10 hours of activity data and analyzed the performance of different recognition algorithms (Chapter 6). For high-level activities such as *going shopping* or *doing housework*, we found that we can achieve recognition rates above 90%, using standard supervised learning techniques such as support vector machines. Our experimental results suggest that it is feasible to recognize such high-level activities using similar techniques as for the recognition of low-level activities.

Unsupervised methods are feasible and valuable for the analysis of high-level activities. While unsupervised methods are important for short-term activities, they become crucial when dealing with long-term and high-level activities, as the cost of annotating such activities is even higher. In Chapter 7, we have introduced an unsupervised approach based on topic models that allows to discover and recognize daily routines such as *working in the office* or *commuting* from body-worn accelerometers. We have evaluated the approach on a data set of more than 80 hours of activity data and shown that it is able to capture to a large extent the structure of the user's daily routines. Using the activity activation patterns as features for a classifier, the approach can predict daily routines in unknown data with high confidence.

8.1 Outlook

In this thesis we have concentrated on two particular challenges for activity recognition, namely reducing the amount of supervision and modeling high-level activities. Towards these goals we have made several contributions. In the next paragraphs, we outline possible future directions into which our work can be continued and extended. Of these we consider the first, namely long-term studies under realistic conditions, particularly important.

Long-term Studies. An important step towards real applications for activity recognition, for which we lacked the time and resources in the course of thesis, is to conduct long-term studies under realistic conditions, for example in cooperation with elderly people in their homes. Such studies would not only give more insight into the feasibility of the different approaches to activity recognition that have been proposed up to now, but also provide valuable information on issues such as user-acceptance, annotation methods, and usability factors. In addition, they would yield useful data for the investigation of long-term and high-level activities, and probably also uncover problems and challenges that researchers aren't even aware of yet. These kind of studies are challenging to set up, and probably require an interdisciplinary approach, e.g. in cooperation with elderly care institutions, smart home facilities or hospitals.

Exploiting topic models further. We believe that the use of topic models as proposed in Chapter 7 is very promising for discovery and modeling of activities, and has interesting properties that we haven't exploited yet. For instance, the probabilistic nature of the approach allows for handling of concurrent and overlapping activities (expressed as co-activation of patterns), and also transitions between activities (e.g. recognizing that the user is "on his way to lunch"). Moreover, the vector of pattern activations for each time step could serve as a high-level feature for more sophisticated classifiers.

Semi-supervised and Active Learning. In Chapter 5 we have investigated the use of semi-supervised learning for activity recognition. Semi-supervised learning has received increasing attention in the machine learning community recently, leading to significant advances in the state of the art [Chapelle *et al.* 2006]. Such methods are appealing, since they allow to learn from large amounts of unlabeled activity data that can be obtained much easier than labeled data, and at the same time allow to combine this data with small but valuable amounts of user feedback. This topic is already being actively explored further, e.g. by [Ali *et al.* 2008]. Another promising approach is active learning, in which the learning algorithm actively asks for labels of informative samples. Since labeling is expensive for activity data, such methods are appealing for the field of activity recognition, and they are already beginning to be explored [Stikic *et al.* 2008b].

Quality of Activities. In various domains, e.g. sports, elderly care, and healthcare, professionals are extremely interested in not only knowing that an activity has been performed, but also how well it has been performed. Recognizing the quality of an activity in an automatic fashion is a challenging problem and an open research question, which could enable a range of interesting applications. E.g., athletes could be provided with personal trainers, patients could get feedback on rehabilitation exercises, and professional care-givers could be eased of the burden to judge their patients performance of ADLs, instead focusing on the care-giving part of their job.

Additional Sensors for Recognition of High-Level Activities. This thesis has mainly investigated the use of acceleration sensors for activity recognition. This type of sensor has many advantages, such as being versatile, well-understood, small, lightweight, and proven to lead to good recognition results for many types of physical activities. However, especially when moving towards high-level activities, other sources of information can be helpful. As an example, we have already incorporated time of day as a feature in the approach described in Chapter 7. We believe the approach presented in this chapter is very versatile, and can easily accommodate for additional and complementary types of information. For instance, location is another important piece of context information, and would be very interesting to investigate in how far it can improve our results. Other possible sources of information include calendar entries or log-files from mobile devices.

List of Figures

List of Tables

Bibliography

[Ali *et al.* 2008] Aziah Ali, Rachel C. King, and Guang Zhong Yang. Semi-supervised Segmentation for Activity Recognition with Multiple Eigenspaces. In *Body Sensor Networks*, 2008. *cited on pp.* 107

[Amft *et al.* 2005] O. Amft, M. Stäger, P. Lukowicz, and G. Tröster. Analysis of chewing sounds for dietary monitoring. In *Proc. 7th Int. Conf. Ubiquitous Computing - UbiComp 2005*, volume 3660, pages 56–72. Springer, 2005. *cited on pp.* 15

[Amft *et al.* 2007] Oliver Amft, Clemens Lombriser, Thomas Stiefmeier, and Gerhard Tröster. Recognition of user activity sequences using distributed event detection. In *Second European Conference on Smart Sensing and Context (EuroSSC)*, October 2007. *cited on pp.* 17, 19

[Andrew *et al.* 2007] Adrienne Andrew, Yaw Anokwa, Karl Koscher, Jonathan Lester, and Gaetano Borriello. Context to make you more aware. *IWSAWC 2007 - The 7th International Workshop on Smart Appliances and Wearable Computing*, 2007. *cited on pp.* 11

[Anliker *et al.* 2004] U. Anliker, J.A. Ward, P. Lukowicz, G. Tröster, F. Dolveck, M. Baer, F. Keita, E.B. Schenker, F. Catarsi, L. Coluccini, A. Belardinelli, D. Shklarski, M. Alon, E. Hirt, R. Schmid, and M. Vuskovic. Amon: a wearable multiparameter medical monitoring and alert system. *IEEE Transactions on Information Technology in Biomedicine*, 8(4):415–427, December 2004. *cited on pp.* 11

[Apple Inc. 2008] Apple Inc. The Apple iPhone [online]. 2008. Available from: `http://www.apple.com/iphone/` [cited May 29, 2008]. *cited on pp.* 12

[Aylward *et al.* 2006] R.P. Aylward, Massachusetts Institute of Technology, Dept. of Architecture, Program in Media Arts, and Sciences. *Sensemble: A Wireless Inertial Sensor System for the Interactive Dance and Collective Motion Analysis*. PhD thesis, Massachusetts Institute of Technology, School of Architecture and Planning, Program in Media Arts and Sciences, 2006. *cited on pp.* 12

[Backman *et al.* 2006] A. Backman, K. Bodin, G. Bucht, LE Janlert, M. Maxhall, T. Pederson, D. Sjölie, B. Sondell, and D. Surie. easyadl - wearable support system for independent life despite dementia. In *CHI 2006 Workshop on Designing Technology for People with Cognitive Impairments*, Montreal, Canada, April 2006. *cited on pp.* 11

[Bao and Intille 2004] L. Bao and S. Intille. Activity recognition from user-annotated acceleration data. In *Proc. Pervasive*, pages 1–17, Vienna, Austria, April 2004. *cited on pp.* 13, 14, 18, 25, 60, 72, 78

[Bardram and Christensen 2007] Jakob E. Bardram and Henrik B. Christensen. Pervasive computing support for hospitals: An overview of the activity-based computing project. *IEEE Pervasive Computing*, 6(1):44–51, 2007. *cited on pp.* 12

[Barry *et al.* 2005] M. Barry, J. Gutknecht, I. Kulka, P. Lukowicz, and T. Stricker. From motion to emotion: a wearable system for the multimedial enhancement of a butoh dance performance. *Journal of Mobile Multimedia*, 1(2):112–132, 2005. *cited on pp.* 12

[Beaudin *et al.* 2007] Jennifer S. Beaudin, Stephen S. Intille, Emmanuel Munguia Tapia, Randy Rockinson, and Margeret E. Morris. Context-Sensitive Microlearning of Foreign Language Vocabulary on a Mobile Device, November 2007. *cited on pp.* 13

[Begole *et al.* 2003] J.B. Begole, J.C. Tang, and R. Hill. Rhythm Modeling, Visualizations, and Applications. *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2003)*, pages 11–20, 2003. *cited on pp.* 17

[Benbasat and Paradiso 2001] A.Y. Benbasat and J.A. Paradiso. An Inertial Measurement Framework for Gesture Recognition and Applications. *Gesture and Sign Language in Human-Computer Interaction, International Gesture Workshop, GW*, 2001. *cited on pp.* 15

[Blei *et al.* 2003] D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003. *cited on pp.* 88, 90

[Blei 2006] David Blei. C implementation of variational EM for latent Dirichlet allocation (LDA), available at http://www.cs.princeton.edu/ blei/lda-c/, 2006. *cited on pp.* 90

[Bobick 1997] A.F. Bobick. Movement, activity and action: the role of knowledge in the perception of motion. *Philosophical Transactions: Biological Sciences*, 352(1358):1257–1265, 1997. *cited on pp.* 15

[Bourke *et al.* 2007] AK Bourke, JV O'Brien, and GM Lyons. Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. *Gait & Posture*, 26(2):194–199, 2007. *cited on pp.* 11

[Brady *et al.* 2005] S. Brady, L.E. Dunne, R. Tynan, D. Diamond, B. Smyth, and G. MP. Garment-Based Monitoring of Respiration Rate Using a Foam Pressure Sensor. *Proceedings of the Ninth IEEE International Symposium on Wearable Computers (ISWC'05)-Volume 00*, pages 214–215, 2005. *cited on pp.* 13

[Brashear *et al.* 2003] Helene Brashear, Thad Starner, Paul Lukowicz, and Holger Junker. Using multiple sensors for mobile sign language recognition. In *ISWC*, pages 45–52, 2003. *cited on pp.* 11

[Buechley and Eisenberg 2007] L. Buechley and M. Eisenberg. Fabric PCBs, electronic sequins, and socket buttons: techniques for e-textile craft. *Personal and Ubiquitous Computing*, pages 1–18, 2007. *cited on pp.* 4

[Buechley 2006] Leah Buechley. A construction kit for electronic textile. In *Proc. of IEEE Int'l Symposium on Wearable Computers (ISWC 2006)*, 2006. *cited on pp.* 5

[Bulling *et al.* 2008] Andreas Bulling, Jamie A. Ward, Hans Gellersen, and Gerhard Tröster. Robust recognition of reading activity in transit using wearable electrooculography. In *Pervasive*, 2008. *cited on pp.* 15

[Cakmakci and Van Laerhoven 2000] O. Cakmakci and K. Van Laerhoven. What shall we teach our pants? In *Wearable Computers, 2000. The Fourth International Symposium on*, 2000. *cited on pp.* 10, 13, 18, 19

[Chambers *et al.* 2002] GS Chambers, S. Venkatesh, GAW West, and HH Bui. Hierarchical recognition of intentional human gestures for sports video annotation. *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, 2, 2002. *cited on pp.* 15

[Chapelle *et al.* 2006] O. Chapelle, B. Schölkopf, and A. Zien. *Semi-supervised learning*. Cambridge, Mass.: MIT Press, 2006. *cited on pp.* 107

[Chen *et al.* 2005] J. Chen, A.H. Kam, J. Zhang, N. Liu, and L. Shue. Bathroom Activity Monitoring Based on Sound. *Proceedings of the International Conference on Pervasive Computing (Pervasive 2005)*, pages 47–61, 2005. *cited on pp.* 15

[Choudhury and Pentland 2003] T. Choudhury and A. Pentland. Sensing and modeling human networks using the sociometer. In *Proc. ISWC*, pages 216–222, October 2003. *cited on pp.* 12, 13, 14

[Choudhury *et al.* 2006] T. Choudhury, M. Philipose, D. Wyatt, and J. Lester. Towards Activity Databases: Using Sensors and Statistical Models to Summarize People's Lives. *IEEE Data Eng. Bull*, 29(1):49–58, 2006. *cited on pp.* 11

[Choudhury *et al.* 2008] Tanzeem Choudhury, Gaetano Borriello, Sunny Consolvo, Dirk Haehnel, Beverly Harrison, Bruce Hemingway, Jeff Hightower, Pedja Klasnja, Karl Koscher, Anthony LaMarca, Jonathan Lester, James Landay, Louis Legrand, Ali Rahimi, Adam Rea, , and Danny Wyatt. The mobile sensing platform: An embedded system for capturing and recognizing human activities. *(submitted to) IEEE Pervasive Computing*, March 2008. *cited on pp.* 5, 14

[Clarkson and Pentland 1998] B. Clarkson and A. Pentland. Extracting context from environmental audio. In *Proceedings of the International Symposium on Wearable Computing*, 1998. *cited on pp.* 9

[Clarkson and Pentland 1999] B. Clarkson and A. Pentland. Unsupervised clustering of ambulatory audio and video. In *icassp*, 1999. *cited on pp.* 16, 19, 73

[Consolvo *et al.* 2008] Sunny Consolvo, D. McDonald, T. Toscos, M. Chen, Jon Froehlich, B. Harrison, P. Klasnja, A. LaMarca, L. LeGrand, R. Libby, I. Smith, and James Landay. Activity Sensing in the Wild: A Field Trial of UbiFit Garden. In *Proceedings of CHI 2008*, 2008. *cited on pp.* 11

[Dey 2001] Anind K. Dey. Understanding and using context. *Personal Ubiquitous Comput.*, 5(1):4–7, February 2001. *cited on pp.* 9

[Duda *et al.* 2004] R. Duda, P. Hart, and D. Stork. *Pattern Recognition*. Wiley and Sons, 2004. *cited on pp.* 54

[Dunne *et al.* 2006] Lucy E. Dunne, Pauline Walsh, Barry Smyth, and Brian Caulfield. Design and evaluation of a wearable optical sensor for monitoring seated spinal posture. In *Proceedings of the 10th IEEE International Symposium on Wearable Computing (ISWC)*, 2006. *cited on pp.* 11, 13

[Eagle and Pentland 2006a] N. Eagle and A. Pentland. Eigenbehaviors: Identifying structure in routine. Technical report, MIT Media Laboratory, 2006. *cited on pp.* 12

[Eagle and Pentland 2006b] N. Eagle and A. Pentland. Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4):255–268, 2006. *cited on pp.* 16

[Enke 2006] Urs Enke. Dansense: Rhythmic analysis of dance movements using acceleration-onset times. Master's thesis, RWTH Aachen University, Aachen, Germany, September 2006. *cited on pp.* 12

[Ermes *et al.* 2008] M. Ermes, J. Pärkkä, J. Mäntyjärvi, and I. Korhonen. Detection of Daily Activities and Sports with Wearable Sensors in Controlled and Uncontrolled Conditions. *IEEE Transactions on Information Technology in Biomedicine*, 12(1):20–26, 2008. *cited on pp.* 15

[Gerasimov 2003] Vadim Gerasimov. *Every Sign of Life*. PhD thesis, Massachusetts Institute of Technology, 2003. *cited on pp.* 13

[Golding and Lesh 1999] A.R. Golding and N. Lesh. Indoor navigation using a diverse set of cheap, wearable sensors. In *Wearable Computers. The Fourth International Symposium on*, 1999. *cited on pp.* 10

[Greenberg and Fitchett 2001] S. Greenberg and C. Fitchett. Phidgets: easy development of physical interfaces through physical widgets. *Proceedings of the 14th annual ACM symposium on User interface software and technology*, pages 209–218, 2001. *cited on pp.* 5

[Hamid *et al.* 2005] R. Hamid, S. Maddi, A. Johnson, A. Bobick, and C. Isbell I. Essa. Unsupervised discovery and characterization of activities from event-streams. In *UAI*, 2005. *cited on pp.* 16

[Hartmann *et al.* 2007] B. Hartmann, L. Abdulla, M. Mittal, and S.R. Klemmer. Authoring sensor-based interactions by demonstration with direct manipulation and pattern recognition. *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 145–154, 2007. *cited on pp.* 5

[Hastie *et al.* 2001] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2001. *cited on pp.* 18

[Heinz *et al.* 2003] E.A. Heinz, K.S. Kunze, S. Sulistyo, I.H. Junker, P. Lukowicz, and G. Troster. Experimental Evaluation of Variations in Primary Features Used for Accelerometric Context Recognition. *Ambient Intelligence: First European Symposium,*

EUSAI 2003, Veldhoven, The Netherlands, November 3-4, 2003: Proceedings, 2003. *cited on pp.* 25

[Heinz *et al.* 2006] E.A. Heinz, KS Kunze, M. Gruber, D. Bannach, and P. Lukowicz. Using wearable sensors for real-time recognition tasks in games of martial arts–An initial experiment. *Proceedings of the 2nd IEEE Symposium on Computational Intelligence and Games (CIG 2006)*, pages 98–102, 2006. *cited on pp.* 12

[Hofmann 2001] Thomas Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning Journal*, 42(1):177–197, 2001. *cited on pp.* 88

[Holmquist *et al.* 2003] L. E. Holmquist, S. Antifakos, B. Schiele, F. Michahelles, M. Beigl, L. Gaye, H. W. Gellersen, A. Schmidt, and M. Strohbach. Building intelligent environments with smart-its. In *Emerging Technologies Exhibition at SIGGRAPH 2003*, San Diego, CA, USA., July 2003. *cited on pp.* 5

[Horvitz *et al.* 2002] Eric Horvitz, Paul Koch, Carl M. Kadie, and Andy Jacobs. Coordinate: Probabilistic Forecasting of Presence and Availability. In *Proc. UAI*, pages 224–233. Morgan Kaufmann Publishers, July 2002. *cited on pp.* 17

[Huỳnh and Schiele 2005] Tâm Huỳnh and Bernt Schiele. Analyzing Features for Activity Recognition. In *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies (sOc-EUSAI)*, pages 159–163, Grenoble, France, 2005. ACM Press New York, NY, USA. *cited on pp.* 6

[Huỳnh and Schiele 2006a] Tâm Huỳnh and Bernt Schiele. Towards Less Supervision in Activity Recognition from Wearable Sensors. In *Proceedings of the 10th IEEE International Symposium on Wearable Computing (ISWC)*, Montreux, Switzerland, October 2006. *cited on pp.* 6, 18, 19, 21, 72

[Huỳnh and Schiele 2006b] Tâm Huỳnh and Bernt Schiele. Unsupervised Discovery of Structure in Activity Data using Multiple Eigenspaces. In *2nd International Workshop on Location- and Context-Awareness (LoCA)*, volume 3987 of *LNCS*, Dublin, Ireland, May 2006. Springer. *cited on pp.* 6, 10, 15, 19, 20, 56

[Huỳnh *et al.* 2007] Tâm Huỳnh, Ulf Blanke, and Bernt Schiele. Scalable Recognition of Daily Activities with Wearable Sensors. In *3rd International Symposium on Location- and Context-Awareness (LoCA)*, pages 50–67, 2007. *cited on pp.* 6, 14, 18, 19, 95

[Huỳnh *et al.* 2008] Tâm Huỳnh, Mario Fritz, and Bernt Schiele. Discovery of Activity Patterns using Topic Models. In *The Tenth International Conference on Ubiquitous Computing (UbiComp)*, 2008. *cited on pp.* 6

[Jaakkola and Haussler 1998] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. *Advances in Neural Information Processing Systems*, 11:487–493, 1998. *cited on pp.* 20, 21, 53

[Jafari *et al.* 2007] R. Jafari, W. Li, R. Bajcsy, S. Glaser, and S. Sastry. Physical Activity Monitoring for Assisted Living at Home. *4th International Workshop on Wearable and Implantable Body Sensor Networks (Bsn 2007): March 26-28, 2007 RWTH Aachen University, Germany*, 2007. *cited on pp.* 11

[Jebara and Pentland 1999] T. Jebara and A. Pentland. Action Reaction Learning: Automatic Visual Analysis and Synthesis of Interactive Behaviour. *Computer Vision Systems: First International Conference, Icvs' 99, Las Palmas, Gran Canaria, Spain, January 13-15, 1999: Proceedings*, 1999. *cited on pp.* 13

[Jebara *et al.* 2004] T. Jebara, R. Kondor, and A. Howard. Probability Product Kernels. *The Journal of Machine Learning Research*, 5:819–844, 2004. *cited on pp.* 53

[Junker *et al.* 2003] H. Junker, Lukowicz P., and G. Tröster. Padnet: wearable physical activity detection network. In *Proc. 7th IEEE Int. Symp. on Wearable Computers - ISWC 2003*, pages 244–245, Washington, DC, USA, 2003. IEEE Computer Society. *cited on pp.* 14

[Junker *et al.* 2004] H. Junker, P. Lukowicz, and J. Mäntyjarvi, editors. *Proceedings of the Benchmarks and a Database for Context Recognition Workshop*, Zurich, Switzerland, April 2004. Part of the Second International Conference on Pervasive Computing (PERVASIVE 2004). *cited on pp.* 4

[Junker *et al.* 2008] Holger Junker, Oliver Amft, Paul Lukowicz, and Gerhard Tröster. Gesture spotting with body-worn inertial sensors to detect user activities. *Pattern Recognition*, 41(6):2010–2024, June 2008. *cited on pp.* 4

[Katz *et al.* 1963] S. Katz, A. Ford, R. Moskowitz, B. Jackson, and M. Jaffe. Studies of Illness in the aged. The Index of ADL: A Standardized Measure of Biological and Psychosocial Function. *JAMA*, 185:914–9, 1963. *cited on pp.* 14

[Kern *et al.* 2003] Nicky Kern, Bernt Schiele, and Albrecht Schmidt. Multi-sensor activity context detection for wearable computing. In *European Symposium on Ambient Intelligence*, Eindhoven, The Netherlands, November 2003. *cited on pp.* 14, 25, 59, 60

[Kern *et al.* 2004] N. Kern, S. Antifakos, B. Schiele, and A. Schwaninger. A model for human interruptability: Experimental evaluation and automatic estimation from wearable sensors. In *Proceedings of the 8th IEEE International Symposium on Wearable Computing (ISWC)*, Washington DC, USA, November 2004. *cited on pp.* 14

[Kern 2005] Nicky Kern. *Multi-Sensor Context-Awareness for Wearable Computing*. PhD thesis, TU Darmstadt, May 2005. *cited on pp.* 72

[Krause *et al.* 2003] A. Krause, DP Siewiorek, A. Smailagic, and J. Farringdon. Unsupervised, dynamic identification of physiological and activity context in wearable computing. *Proceedings of the 7th IEEE International Symposium on Wearable Computers*, pages 88–97, 2003. *cited on pp.* 14, 25, 60

[Krumm and Horvitz 2006] J. Krumm and E. Horvitz. Predestination: Inferring Destinations from Partial Trajectories. In *Proc. UbiComp*, 2006. *cited on pp.* 17

[Kunze *et al.* 2006] K. Kunze, M. Barry, E.A. Heinz, P. Lukowicz, D. Majoe, and J. Gutknecht. Towards Recognizing Tai Chi–An Initial Experiment Using Wearable Sensors. In *4th International Forum on Applied Wearable Computing (IFAWC)*, April 2006. *cited on pp.* 15, 18

[Langheinrich 2005] Marc Langheinrich. *Personal Privacy in Ubiquitous Computing – Tools and System Support*. PhD thesis, ETH Zurich, Zurich, Switzerland, May 2005. *cited on pp.* 5

[Lasserre *et al.* 2006] J.A. Lasserre, C.M. Bishop, and T.P. Minka. Principled hybrids of generative and discriminative models. In *CVPR*, volume 6, pages 87–94, 2006. *cited on pp.* 21

[Lawton and Brody 1969] M.P. Lawton and E.M. Brody. Assessment of older people: Self-maintaining and instrumental activities of daily living – Instrumental Activities of Daily Living Scale (ADL) . In *Gerontologist*, pages 179–186, 1969. *cited on pp.* 15

[Lee and Mase 2002] S.W. Lee and K. Mase. Activity and location recognition using wearable sensors. *Pervasive Computing, IEEE*, 1(3):24–32, 2002. *cited on pp.* 14, 25

[Leonardis *et al.* 2002] A. Leonardis, H. Bischof, and J. Maver. Multiple eigenspaces. *Pattern Recognition*, 35(11):2613–2627, 2002. *cited on pp.* 36, 37, 39, 54

[Lester *et al.* 2005] Jonathan Lester, Tanzeem Choudhury, Nicky Kern, Gaetano Borriello, and Blake Hannford. A hybrid discriminative/generative approach for modeling human activities. In *Proc. IJCAI*, pages 766–772, Edinburgh, United Kingdom, August 2005. *cited on pp.* 19, 21, 24, 53, 73

[Lester *et al.* 2006] Jonathan Lester, Tanzeem Choudhury, and Gaetano Borriello. A practical approach to recognizing physical activities. In *Proc. Pervasive*, May 2006. *cited on pp.* 10

[Liao *et al.* 2007] L. Liao, D. Fox, and H. Kautz. Extracting Places and Activities from GPS Traces Using Hierarchical Conditional Random Fields. *The International Journal of Robotics Research*, 26(1):119, 2007. *cited on pp.* 17, 19

[Linz *et al.* 2006] Torsten Linz, Christine Kallmayer, Rolf Aschenbrenner, and Herbert Reichl. Fully Integrated EKG Shirt based on Embroidered Electrical Interconnections with Conductive Yarn and Miniaturized Flexible Electronics. In *International Workshop on Wearable and Implantable Body Sensor Networks*, 2006. *cited on pp.* 13

[Liszka *et al.* 2004] K. Liszka, Michael A. Mackin, Michael J. Lichter, David W. York, Dilip Pillai, and David S. Rosenbaum. Keeping a Beat on the Heart. *IEEE Pervasive Computing*, 3(4):42–49, 2004. *cited on pp.* 11

[Lo *et al.* 2005] B. Lo, S. Thiemjarus, R. King, and G. Yang. Body Sensor Network–A Wireless Sensor Platform for Pervasive Healthcare Monitoring. In *Proc. Pervasive*, 2005. *cited on pp.* 70

[Logan *et al.* 2007] Beth Logan, Jennifer Healey, Matthai Philipose, Emmanuel Munguia Tapia, and Stephen S. Intille. A Long-Term Evaluation of Sensing Modalities for Activity Recognition. In *Proceedings of UBICOMP 2007: The 9th International Conference on Ubiquitous Computing*, October 2007. *cited on pp.* 14

[Loosli *et al.* 2003] G. Loosli, S. Canu, and A. Rakotomamonjy. Détection des activités quotidiennes à l'aide des séparateurs à Vaste Marge. *RJCIA, France*, pages 139–152, 2003. *cited on pp.* 19

[Lukowicz *et al.* 2004] Paul Lukowicz, Jamie A. Ward, Holger Junker, Mathias Stäger, Gerhard Tröster, Amin Atrash, and Thad Starner. Recognizing workshop activity using body worn microphones and accelerometers. In *Pervasive Computing: Proceedings of the 2nd International Conference*, pages 18–22. Springer-Verlag Heidelberg: Lecture Notes in Computer Science, April 2004. *cited on pp.* 14, 15, 73

[Lukowicz *et al.* 2006] P. Lukowicz, F. Hanser, C. Szubski, and W. Schobersberger. Detecting and interpreting muscle activity with wearable force sensors. In *Proc. 4th International Conference PERVASIVE 2006*, pages 101–116. Springer LNCS, May 2006. *cited on pp.* 13

[Lukowicz *et al.* 2007] P. Lukowicz, A. Timm-Giel, M. Lawo, and O. Herzog. Wearit@work: Toward real-world industrial wearable computing. *IEEE Pervasive Computing*, 6(4):8–13, Oct-Dec 2007. *cited on pp.* 11

[Maitland *et al.* 2006] Julie Maitland, Scott Sherwood, Louise Barkhuus, Ian Anderson, Malcolm Hall, Barry Brown, Matthew Chalmers, and Henk Muller. Increasing the awareness of daily activity levels with pervasive computing. In *1st International Conference on Pervasive Computing Technologies for Healthcare 2006*, 2006. *cited on pp.* 11

[Manning *et al.* 2008] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008. *cited on pp.* 26

[Mantyjarvi *et al.* 2001] J. Mantyjarvi, J. Himberg, T. Seppanen, and N.R. Center. Recognizing human motion with multiple acceleration sensors. In *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, volume 2, 2001. *cited on pp.* 14, 25

[Maurer *et al.* 2006] U. Maurer, A. Smailagic, D.P. Siewiorek, and M. Deisher. Activity Recognition and Monitoring Using Multiple Sensors on Different Body Positions. *International Workshop on Wearable and Implantable Body Sensor Networks (BSN)*, 2006. *cited on pp.* 18

[Mayrhofer and Gellersen 2007] R. Mayrhofer and Hans Gellersen. Shake well before use: Authentication based on accelerometer data. In *Pervasive*, 2007. *cited on pp.* 5

[Mikolajczyk *et al.* 2005] K. Mikolajczyk, B. Leibe, and B. Schiele. Local features for object class recognition. *Proc. ICCV*, 2:1792–1799, 2005. *cited on pp.* 24

[Minnen *et al.* 2006a] D. Minnen, T. Westeyn, T. Starner, J. Ward, and P. Lukowicz. Performance Metrics and Evaluation Issues for Continuous Activity Recognition. *Performance Metrics for Intelligent Systems*, 2006. *cited on pp.* 4

[Minnen *et al.* 2006b] David Minnen, Thad Starner, Irfan Essa, and Charles Isbell. Discovering characteristic actions from on-body sensor data. In *Proc. ISWC*, October 2006. *cited on pp.* 10, 15, 16, 19, 72

[Minnen *et al.* 2007] D. Minnen, T. Westeyn, D. Ashbrook, P. Presti, and T. Starner. Recognizing Soldier Activities in the Field. *4th International Workshop on Wearable and*

Implantable Body Sensor Networks (Bsn 2007): March 26-28, 2007 RWTH Aachen University, Germany, 2007. *cited on pp.* 13, 19

[Montoye *et al.* 1983] H.J. Montoye, R. Washburn, Servais S., A. Ertl, J.G. Webster, and F.J. Nagle. Estimation of energy expenditure by a portable accelerometer. *Medicine & Science in Sports & Exercise*, 15(5):403, 1983. *cited on pp.* 10

[Naeem *et al.* 2007] Usman Naeem, John Bigham, and Jinfu Wang. Recognising Activities of Daily Life using Hierarchical Plans. In *EuroSSC*, October 2007. *cited on pp.* 13

[Ng and Jordan 2002] A. Ng and M. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing Systems*. MIT Press, 2002. *cited on pp.* 20

[Nilsback and Caputo 2004] ME Nilsback and B. Caputo. Cue integration through discriminative accumulation. *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, 2, 2004. *cited on pp.* 20

[Nintendo 2008] Nintendo. Wii Entertainment System [online]. May 2008. Available from: http://www.nintendo.com/wii [cited May 29, 2008]. *cited on pp.* 12

[Nowozin *et al.* 2007] S. Nowozin, G. Bakir, and K. Tsuda. Discriminative Subsequence Mining for Action Classification. *IEEE International Conference on Computer Vision (ICCV)*, 2007. *cited on pp.* 13

[Oliver and Flores-Mangas 2006] N. Oliver and F. Flores-Mangas. HealthGear: A Real-time Wearable System for Monitoring and Analyzing Physiological Signals. In *Proc. Body Sensor Networks*, pages 61–64, 2006. *cited on pp.* 13

[Oliver *et al.* 2002] N. Oliver, E. Horvitz, and A. Garg. Layered representations for human activity recognition. *Proc. ICMI*, 2002. *cited on pp.* 17, 73, 95

[Paradiso *et al.* 2005] R. Paradiso, G. Loriga, and N. Taccini. A wearable health care system based on knitted integrated sensors. *Information Technology in Biomedicine, IEEE Transactions on*, 9(3):337–344, 2005. *cited on pp.* 11

[Patterson *et al.* 2004] Donald J. Patterson, Lin Liao, Krzysztof Gajos, Michael Collier, Nik Livic, Katherine Olson, Shiaokai Wang, Dieter Fox, and Henry Kautz. Opportunity Knocks: a System to Provide Cognitive Assistance with Transportation Services. In Itiro Siio Nigel Davies, Elizabeth Mynatt, editor, *Proceedings of UBICOMP 2004: The Sixth International Conference on Ubiquitous Computing*, volume LNCS 3205, pages 433–450. Springer-Verlag, October 2004. *cited on pp.* 11, 19

[Patterson *et al.* 2005] DJ Patterson, D. Fox, H. Kautz, and M. Philipose. Fine-grained activity recognition by aggregating abstract object usage. In *Proc. ISWC*, pages 44–51, 2005. *cited on pp.* 13, 19

[Pentland *et al.* 2005] A. Pentland, T. Choudhury, N. Eagle, and P. Singh. Human dynamics: computation for organizations. *Pattern Recognition Letters*, 26(4):503–511, 2005. *cited on pp.* 12

[Pentland 2007] A. Pentland. Automatic mapping and modeling of human networks. *Physica A: Statistical Mechanics and its Applications*, 378(1):59–67, 2007. *cited on pp.* 12

[Perreira *et al.* 2004] F. Perreira, A. Bernal, K. Crammer, and R. McDonald. Linear models for structure prediction. In *NIPS Workshop on Graphical Models and Kernels*, 2004. *cited on pp.* 21

[Philipose *et al.* 2004] Matthai Philipose, Kenneth P. Fishkin, Mike Perkowitz, Donald J Patterson Dirk Hahnel, Dieter Fox, and Henry Kautz. Inferring Activities from Interactions with Objects. *IEEE Pervasive Computing: Mobile and Ubiquitous Systems*, 3(4):50–57, 2004. *cited on pp.* 13, 15, 19

[Randell and Muller 2000] Cliff Randell and Henk Muller. Context awareness by analysing accelerometer data. In Blair MacIntyre and Bob Iannucci, editors, *The Fourth International Symposium on Wearable Computers*, pages 175–176. IEEE Computer Society, October 2000. *cited on pp.* 10

[Ravi *et al.* 2005] N. Ravi, N. Dandekar, P. Mysore, and M. Littman. Activity recognition from accelerometer data. In *Proceedings of the Seventeenth Conference on Innovative Applications of Artificial Intelligence (IAAI)*, 2005. *cited on pp.* 14, 18, 19, 25, 60, 72

[Rhodes 1997] B.J. Rhodes. The wearable remembrance agent: A system for augmented memory. *Personal and Ubiquitous Computing*, 1(4):218–224, 1997. *cited on pp.* 10

[Sala *et al.* 2007] Matthias C. Sala, Kurt Partridge, Linda Jacobson, and James Begole. An exploration into activity-informed physical advertising using pest. In *Pervasive*, pages 73–90, 2007. *cited on pp.* 13

[Salber *et al.* 1999] Daniel Salber, Anind K. Dey, and Gregory D. Abowd. The context toolkit: Aiding the development of context-enabled applications. In *CHI*, pages 434–441, 1999. *cited on pp.* 5

[Schiele *et al.* 1999] Bernt Schiele, Nuria Oliver, Tony Jebara, and Alex Pentland. An Interactive Computer Vision System - DyPERS: Dynamic Personal Enhanced Reality System. In *ICVS'99 International Conference on Vision Systems*, 1999. *cited on pp.* 9

[Shi *et al.* 2004] Y. Shi, Y. Huang, D. Minnen, A. Bobick, and I. Essa. Propagation networks for recognition of partially ordered sequential action. *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, 2, 2004. *cited on pp.* 13

[Si *et al.* 2007] Hua Si, Seung Jin Kim, Nao Kawanishi, and Hiroyuki Morikawa. A context-aware reminding system for daily activities of dementia patients. In *ISWAWC*, page 50, 2007. *cited on pp.* 11

[Stäger *et al.* 2004] M. Stäger, P. Lukowicz, and G. Tröster. Implementation and evaluation of a low-power sound-based user activity recognition system. In *Proc. 8th Int. Symp. On Wearable Computers - ISWC 2004*, pages 138–141. IEEE Computer Society, 2004. *cited on pp.* 13

[Stanford 2002] V. Stanford. Wearable Computing Goes Live in Industry. *IEEE Pervasive Computing*, 1(4):14–19, 2002. *cited on pp.* 11

[Starner *et al.* 1997] T. Starner, J. Weaver, and A. Pentland. A Wearable Computer-based American Sign Language Recogniser. *Personal and Ubiquitous Computing*, 1(4):241–250, 1997. *cited on pp.* 9

[Starner *et al.* 1999] Thad Starner, Bradley Rhodes, Joshua Weaver, and Alex Pentland. Everyday-use Wearable Computers. In *International Symposium on Wearable Computers*, 1999. *cited on pp.* 9

[Stiefmeier *et al.* 2006] Thomas Stiefmeier, Georg Ogris, Holger Junker, Paul Lukowicz, and Gerhard Tröster. Combining motion sensors and ultrasonic hands tracking for continuous activity recognition in a maintenance scenario. In *Proc. ISWC*, 2006. *cited on pp.* 10, 14, 15

[Stiefmeier *et al.* 2007] Thomas Stiefmeier, Daniel Roggen, and Gerhard Tröster. Gestures are Strings: Efficient Online Gesture Spotting and Classification using String Matching. In *BodyNets*, June 2007. *cited on pp.* 15

[Stiefmeier *et al.* 2008] T. Stiefmeier, D. Roggen, G. Ogris, P. Lukowicz, and G. Tröster. Wearable Activity Tracking in Car Manufacturing. *IEEE Pervasive Computing*, 7(2), April-June 2008. *cited on pp.* 12, 15, 19

[Stikic *et al.* 2008a] M. Stikic, Tâm Huỳnh, K. Van Laerhoven, and B. Schiele. ADL Recognition Based on the Combination of RFID and Accelerometer Sensing. In *2nd International Conference on Pervasive Computing Technologies for Healthcare 2008*, 2008. *cited on pp.* 6, 14, 15, 19

[Stikic *et al.* 2008b] Maja Stikic, Kristof Van Laerhoven, and Bernt Schiele. Exploring Semi-Supervised and Active Learning for Activity Recognition. In *International Symposium on Wearable Computers (ISWC)*, 2008. *cited on pp.* 14, 19, 107

[Subramanya *et al.* 2006] A. Subramanya, A. Raj, J. Bilmes, and D. Fox. Recognizing activities and spatial context using wearable sensors. In *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI)*, Cambridge, MA, USA, July 2006. *cited on pp.* 14, 19

[Tapia and Intille 2007] Emmanuel Munguia Tapia and Stephen Intille. Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In *International Symposium on Wearable Computers (ISWC)*, 2007. *cited on pp.* 15, 18

[Tapia *et al.* 2004] Emmanuel Munguia Tapia, Stephen S. Intille, and Kent Larson. Activity recognition in the home using simple and ubiquitous sensors. In *Pervasive*, pages 158–175, 2004. *cited on pp.* 15

[Tentori and Favela 2008] M. Tentori and J. Favela. Activity-Aware Computing for Healthcare. *IEEE PERVASIVE COMPUTING*, pages 51–57, 2008. *cited on pp.* 12

[Torralba *et al.* 2004] A. Torralba, KP Murphy, and WT Freeman. Sharing features: efficient boosting procedures for multiclass object detection. *Computer Vision and Pattern*

Recognition, Proceedings of the 2004 IEEE Computer Society Conference on, 2, 2004. *cited on pp.* 20

[Tsuda *et al.* 2002] K. Tsuda, M. Kawanabe, G. Rätsch, S. Sonnenburg, and K.-R. Müller. A new discriminative kernel from probabilistic models. *Neural Computation*, 14(10):2397–2414, 2002. *cited on pp.* 53

[Van Laerhoven and Gellersen 2004] Kristof Van Laerhoven and Hans-Werner Gellersen. Spine versus porcupine: A study in distributed wearable activity recognition. In *Proc. ISWC*, Washington DC, USA, 2004. *cited on pp.* 13, 14, 25

[Van Laerhoven *et al.* 2003] K. Van Laerhoven, N. Kern, H.W. Gellersen, and B. Schiele. Towards a wearable inertial sensor network. *In Proc. of the IEE Eurowearable*, pages 125–130, 2003. *cited on pp.* 18, 60

[Van Laerhoven *et al.* 2006] K. Van Laerhoven, H.W. Gellersen, and Y.G. Malliaris. Long-Term Activity Monitoring with a Wearable Sensor Node. *BSN Workshop*, 2006. *cited on pp.* 4, 5, 69, 90

[Van Laerhoven *et al.* 2008a] Kristof Van Laerhoven, Marko Borazio, and Bernt Schiele. Sustained Logging and Discrimination of Sleep Postures with Low-Level, Wrist-Worn Sensors. In *International Symposium on Wearable Computers (ISWC)*, 2008. *cited on pp.* 17

[Van Laerhoven *et al.* 2008b] Kristof Van Laerhoven, David Kilian, and Bernt Schiele. Using Rhythm Awareness in Sustained Activity Recognition. In *International Symposium on Wearable Computers (ISWC)*, 2008. *cited on pp.* 17

[Van Laerhoven 2004] Kristof Van Laerhoven. Medical healthcare monitoring with wearable and implantable sensors. In *UbiHealth 2004: The 3rd International Workshop on Ubiquitous Computing for Pervasive Healthcare Applications*, 2004. *cited on pp.* 11

[Van Laerhoven 2005] Kristof Van Laerhoven. *Embedded Perception*. PhD thesis, Lancester University, 2005. *cited on pp.* 19

[Van Laerhoven 2007] K. Van Laerhoven. Memorizing what you did last week: Towards detailed actigraphy with a wearable sensor. In *IWSAWC 2007 - The 7th International Workshop on Smart Appliances and Wearable Computing*, 2007. *cited on pp.* 17

[Vapnik 1998] V. Vapnik. *Statistical learning theory*. Wiley and Sons, NY, 1998. *cited on pp.* 56, 57

[Vasconcelos *et al.* 2004] N. Vasconcelos, P. Ho, and P. J. Moreno. The kullback-leibler kernel as a framework for discriminant and localized representations for visual recognition. In *European Conference on Computer Vision*, pages 430–441. Springer, 2004. *cited on pp.* 53

[Villalba *et al.* 2006] E. Villalba, M. Ottaviano, M.T. Arredondo, A. Martinez, and S. Guillen. Wearable monitoring system for heart failure assessment in a mobile environment. In *Computers in Cardiology*, volume 33, 2006. *cited on pp.* 11

[Wang *et al.* 2007] S. Wang, W. Pentney, A.M. Popescu, T. Choudhury, and M. Philipose. Common Sense Based Joint Training of Human Activity Recognizers. In *Proc. IJCAI*, 2007. *cited on pp.* 10, 14, 15, 19

[Ward *et al.* 2005] J.A. Ward, P. Lukowicz, and G. Tröster. Gesture spotting using wrist worn microphone and 3-axis accelerometer. *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence*, pages 99–104, 2005. *cited on pp.* 15

[Ward *et al.* 2006a] J.A. Ward, Lukowicz, Tröster P., and T. G., Starner. Activity recognition of assembly tasks using body-worn microphones and accelerometers. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28:10:1553–1567, October 2006. *cited on pp.* 12, 15, 19

[Ward *et al.* 2006b] Jamie A. Ward, Paul Lukowicz, and Gerhard Tröster. Evaluating performance in continuous context recognition using event-driven error characterisation. In *LoCA*, pages 239–255, 2006. *cited on pp.* 4

[Weiser 1991] Mark Weiser. The Computer for the 21st Century. *Scientific American*, 265(3):94–104, September 1991. *cited on pp.* 1, 9

[Westeyn *et al.* 2006] T. Westeyn, P. Presti, and T. Starner. ActionGSR: A combination galvanic skin response-accelerometer for physiological measurements in active environments. *Proc 10th IEEE Int Symp on Wearable Computers, Montreux, Switzerland*, pages 129–130, 2006. *cited on pp.* 13

[Wong *et al.* 1981] T.C. Wong, J.G. Webster, H.J. Montoye, and R. Washburn. Portable Accelerometer Device for Measuring Human Energy Expenditure. *Biomedical Engineering, IEEE Transactions on*, pages 467–471, 1981. *cited on pp.* 10

[Wren *et al.* 2007] Christopher R. Wren, Yuri A. Ivanov, Ishwinder Kaur, Darren Leigh, and Jonathan Westhues. Socialmotion: Measuring the hidden social life of a building. In *LoCA*. Springer, 2007. *cited on pp.* 4

[Wyatt *et al.* 2005] D. Wyatt, M. Philipose, and T. Choudhury. Unsupervised Activity Recognition Using Automatically Mined Common Sense. *Proc. AAAI 2005*, 2005. *cited on pp.* 10

[Xybernaut Corp. 2008] Xybernaut Corp. [online]. 2008. Available from: `http://www.xybernaut.com/` [cited May 29, 2008]. *cited on pp.* 11

[Zacks and Tversky 2001] JM Zacks and B. Tversky. Event structure in perception and conception. *Psychological Bulletin*, 127(1), 2001. *cited on pp.* 86

[Zhang and Hartmann 2007] H. Zhang and B. Hartmann. Building upon everyday play. In *Conference on Human Factors in Computing Systems (CHI)*. ACM Press New York, NY, USA, 2007. *cited on pp.* 12

[Zinnen and Schiele 2008] Andreas Zinnen and Bernt Schiele. A new Approach to Enable Gesture Recognition in Continuous Data Streams. In *International Symposium on Wearable Computers (ISWC)*, 2008. *cited on pp.* 4

[Zinnen *et al.* 2007] A. Zinnen, K. Van Laerhoven, and B. Schiele. Toward Recognition of Short and Non-repetitive Activities from Wearable Sensors. In *European Conference on Ambient Intelligence*. Springer, 2007. *cited on pp.* 15

LEBENSLAUF

Tâm Huynh

Geboren am 29. Juli 1975 in Paris, Frankreich.
Nationalität: deutsch
Familienstand: verheiratet

AUSBILDUNG

1997 – 2003	*Technische Universität Darmstadt* *Studiengang Informatik* Diplom, 07/2003 Diplomarbeit: *Konzeption und Implementierung einer skizzenbasierten Benutzungsschnittstelle im Kontext von Geoinformationssystemen* Gesamturteil ‚sehr gut' Vordiplom, 09/1999 Gesamturteil ‚sehr gut'
09/2000 – 04/2001	*University of British Columbia, Vancouver, Canada* *Department of Computer Science* Graduate Studies Im Rahmen eines Austauschprogramms mit der TU Darmstadt
1996 - 1997	*Technische Universität Darmstadt* *Studiengang Architektur*
1995 - 1996	*Friedrich Fröbel Schule, Maintal* Zivildienst
1988 - 1995	*Karl Rehbein Gymnasium, Hanau* Abitur (Note 1,0)

STIPENDIEN

2000 - 2001	Stipendiat des DAAD während eines Studienaufenthaltes an der University of British Columbia, Kanada

BESCHÄFTIGUNG

seit 11/2004 — *Technische Universität Darmstadt, FB Informatik*
Wissenschaftlicher Mitarbeiter

Als Mitarbeiter in der Gruppe „Multimodale Interaktive Systeme“ von Prof. Schiele entwickele ich Methoden zur Kontexterkennung mittels tragbarer Sensoren. Anwendungsgebiete hierfür finden sich z.B. im Bereich der Mensch-Computer-Interaktion, oder in neuen Technologien für die alternde Gesellschaft. Abschluss voraussichtlich 8/2008 mit Promotion.

08/2003 – 10/2004 — *nterra e-strategy and consulting GmbH, Griesheim*
Entwickler und Consultant

Meine Schwerpunkte lagen in der Integration von Unternehmensanwendungen mittels EAI-Plattformen sowie Entwicklung von Web-Applikationen mit dem J2EE-Framework.

BESCHÄFTIGUNG WÄHREND DES STUDIUMS

2001 – 2002 — *Fraunhofer Institut für Graphische Datenverarbeitung*
Wissenschaftliche Hilfskraft

Abteilung industrielle Applikationen (A5), betreut von Paul Benölken. Thema: 3D-Visualisierung von Meßdaten (C++, Open Inventor auf HP-UX und Win2000).

INDUSTRIEPROJEKTE

IM RAHMEN MEINER TÄTIGKEIT ALS ENTWICKLER & CONSULTANT, NTERRA GMBH, GRIESHEIM

04/2003 – 09/2004 — *Verschiedene Projekte im Bereich EAI, Web-Entwicklung und Internet-Portale, u.a. in Zusammenarbeit mit den Firmen BASF, Bosch, ContiTeves, SupplyOn, LoyaltyPartner. Detailliertere (anonymisierte) Beschreibungen liefere ich gerne nach.*

Aufgaben: Entwurf und Implementierung von EAI-Logik, Web-Front- und –Backends, Portal-Komponenten, SAP-Anbindungen
Technologien: webMethods EAI Plattform, Java, sowie eine Vielzahl von Web-Technologien.

WISSENSCHAFTLICHE PROJEKTE

TU DARMSTADT, FB INFORMATIK

05/2005 – 08/2008 — *MOBVIS - Vision Technologies and Intelligent Maps for mobile Interfaces in Urban Scenarios (European Commisson FP6-511051)*

Partner: TU Darmstadt, KTH Stockholm, Univ. of Ljubljana, TeleAtlas, Joanneum Research Graz
Aufgaben: Unterstützung der Orientierung und Navigation in urbanen Umgebungen durch Kontext-Erkennung mit tragbaren Sensoren
Technologien: u.a. Xsens Mtx inertial sensors, Matlab, Java, Webservices
Weitere Infos: www.mobvis.org

05/2005 –08/2008 **CESORA - Context Environment for Service-Oriented Business Applications**

Partner: TU Darmstadt, SAP Research Darmstadt
Aufgaben: Analyse und Design von Applikations-Szenarien für Kontext-bezogene Dienste

LEHRTÄTIGKEIT

TU DARMSTADT, FB INFORMATIK

WS 2007/2008 *Software-Engineering Praktikum zum Thema „Human Computer Systems"*
Betreuung von drei Gruppen zum Thema „Eine Display-Software für Sportveranstaltungen"

WS 2006/2007 *Vorlesung „Grundlagen der Informatik I" (600+ Studenten), Prof. Schiele*
Themen: objektorientierte und funktionale Programmierung (Java, Scheme)
Konzeption und Organisation der vorlesungsbegleitenden Übungen und Klausuren

WS 2005/2006 *Seminar „Wireless Sensor Networks"*
Betreuung einer Arbeit zum Thema „Activity Recognition with Wearable Sensors"

WS 2005/2006 *Vorlesung „Grundlagen der Informatik I" (500+ Studenten), Prof. Mezini*
Themen: objektorientierte und funktionale Programmierung (Java, Scheme)
Konzeption und Organisation der vorlesungsbegleitenden Übungen und Klausuren

SS 2005 *Software-Engineering Praktikum zum Thema „Human Computer Systems"*
Betreuung von drei Gruppen zum Thema „An Annotation Tool for Multimodal Sensor Data"

UNIVERSITY OF BRITISH COLUMBIA, DEPARTMENT OF COMPUTER SCIENCE

2000/2001 *CPSC 319 Software Engineering Project: „ The design, implementation, and test of a large software system, using a team approach.", Anne Lavergne*
Betreuung und Korrektur der vorlesungsbegleitenden Übungen, sowie Betreuung von zwei studentischen Projekt-Teams über zwei Semester.

NTERRA GMBH, GRIESHEIM

06/2004 *Workshop „SAP Business Connector", supplyOn AG, Halbergmoos*
Vorbereitung und Durchführung eines 1-tägigen Workshops inkl. Übungen

BETREUTE BACHELOR- UND DIPLOMARBEITEN

TU DARMSTADT, FB INFORMATIK

SS 2006 Lars Ax. *Localization using Heterogeneous Sensors,* Diplomarbeit

SS 2006 Ulf Blanke, *Unsupervised Activity Recognition using Wearable Sensors,* Diplomarbeit

WS 2005/2006 Marcus Rohrbach, *An Annotation Tool for Multimodal Sensor Data,* Bachelorarbeit

SS 2005 Martin Müller, *Location Estimation using Wireless LAN Beacons,* Diplomarbeit

SS 2005 Ulrich Steinhoff, *Location Estimation using Wearable Sensors,* Diplomarbeit

VERÖFFENTLICHUNGEN

KONFERENZEN

Tâm Huynh, Mario Fritz and Bernt Schiele. *Discovery of Activity Patterns using Topic Models*. To Appear in Tenth International Conference on Ubiquitous Computing (UbiComp 2008), Sep. 21-24, 2008, Seoul, South Korea.

Maja Stikic, Tâm Huynh, Kristof Van Laerhoven and Bernt Schiele. *ADL Recognition Based on the Combination of RFID and Accelerometer Sensing*. 2nd International Conference on Pervasive Computing Technologies for Healthcare 2008, Tampere, Finland.

Tâm Huynh, Ulf Blanke and Bernt Schiele. *Scalable Recognition of Daily Activities from Wearable Sensors*. 3rd International Symposium on Location- and Context-Awareness (LoCA), September 2007, Oberpfaffenhofen, Germany. Springer.

Tâm Huynh and Bernt Schiele. *Towards Less Supervision in Activity Recognition from Wearable Sensors*. Proceedings of the 10th IEEE International Symposium on Wearable Computing (ISWC). October 2006, Montreux, Switzerland.

Tâm Huynh and Bernt Schiele. *Unsupervised discovery of structure in activity data using multiple eigenspaces*. 2nd International Workshop on Location- and Context-Awareness (LoCA), May 2006, Dublin, Ireland.

Tâm Huynh and Bernt Schiele. Analyzing *Features for Activity Recognition*. Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies, October 2005, Grenoble, France, ACM Press New York, NY, USA.

VORTRÄGE

Tâm Huynh. *What do my accelerometers tell me about my day?*, Vortrag im Rahmen des Dagstuhl Seminars: „Mobile Interfaces Meet Cognitive Technologies“ (September 2007), Dagstuhl Seminar Proceedings 07371, Leibniz-Zentrum für Informatik, Schloss Dagstuhl, Wadern

SPRACHKENNTNISSE

Deutsch	Muttersprache
Englisch	Sehr gut in Wort und Schrift (9 Jahre Schule, 1 Jahr Kanada, 4 Jahre als Arbeitssprache)
Französisch	Gut in Wort und Schrift (7 Jahre Schule, franz. Verwandschaft)
Vietnamesisch	Grundkenntnisse in Wort (Verständnis und passiver Wortschatz durch Vater)
Japanisch	Grundkenntnisse in Wort und Schrift (5 Semester Sprachkurs, TU Darmstadt)
Chinesisch	Grundkenntnisse in Wort und Schrit (VHS-Kurs, chin. Verwandte)

IT KENNTNISSE

PROGRAMMIERUNG

Java	Industrieprojekte u.a. mit: Servlets, Swing, Applets, JNI, JAXP, JDBC, JUnit, JavaMail, Webservices, Jakarta Turbine, Velocity (MVC Framework), Jakarta Jetspeed (Portal Framework), Apache Ant, Apache Axis, SAP JCo (SAP R/3 – Java Schnittstelle)
Web	Industrieprojekte u.a. mit : XML, XSL, HTML, CSS, Javascript
EAI	Flow (webMethods) (mehrere Industrieprojekte), ABAP (Grundkenntnisse)
C, C++	OpenGL (Grundkenntnisse) OpenInventor (Grundkenntnisse)
Sonstige	Matlab (mehrjährige Erfahrung) Scheme, Eiffel, Pascal

ANWENDUNGS-SOFTWARE

EAI-Plattformen	Industrieprojekte mit webMethods Integration Server, webMethods Developer / Trading Networks SAP Business Connector, inubit Integration Server
Datenbanken	Microsoft SQL Server, mySQL
Web / Portale	Jakarta Tomcat, Apache Jetspeed, iPlanet LDAP Server, Apache Webserver, GNU Mailman
Betriebssysteme	Anwendung und Administration von Windows 9x/2000/XP/Vista, Anwendung von Mac OSX & diversen GNU/Linux Distributionen, Grundkenntnisse Unix-Shell Scripting
Sonstige	MS-Office Suite, LaTeX

Datum Unterschrift