Development of a Smartphone Based System to Support Personal Journaling of Daily Activities

Dissertation
zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
der Technischen Fakultät der Universität Bielefeld

vorgelegt von Sebastian Hammerl am 16.04.2013

Acknowledgements

During the last three and a half years many colleagues, friends, and others supported me while writing this thesis. I would like to thank everybody who helped me in realizing this work, thank you very much!

In particular I thank my supervisors Thomas Hermann and Helge Ritter for their assistance and lively discussions. Many ideas were formed in meetings and brain storming sessions which finally led to the topic and this thesis. They gave me the liberty to work freely and yet were always available for questions. The same applies to my office colleagues Christian Leichsenring, Rene Tünnermann and Ulf Großekathöfer. In endless inspiring discussions they helped me to think outside the box, solve problems, and develop ideas. I would also like to mention all remaining group members and former group members of the *Ambient Intelligence Group*: Till Bovermann, Florian Grond, Tobias Großhauser, Bastian Kriesten, Alexander Neumann, Eckard Riedenklau, and Sebastian Zehe. The group and all the fantastic members offered me a great environment every day. It has been a pleasure to work with you! I also would like to thank the *Neuroinformatics Group* with its great members, who were always available for any kind of question. I do not want to forget Tobias Rodehutskors, who worked for me as a student assistant on the implementation of the AMARAS system. His programming skills and autonomous working style enabled me to concentrate on my thesis. My thanks also goes to the personnel of the CITEC, especially Claudia Muhl, who guided me through all the administrative stuff which I had to do.

I thank my parents for supporting and motivating me. They are always there for me and without them I could not have done this work. They paved the way through my whole life and enabled me to reach this goal. I also thank my brother, sister, and my grandparents. My whole family believes in me and I cannot imagine better people around me. Aside from work and family, I would like to thank all my friends at my scout group from Lage, *Stamm der Ritter, Pfadfindergruppe Lage (Lippe) e. V.*[1]. Whenever I needed some distance to my theoretical work I had a place there to hang out with friends, do some real physical work, and recharge my energy for the work on my thesis. Especially the *Grizzlies* and Fiete were always available to renovate our house and build the future for the youth group. Finally, I thank Amelie Beckmann for her love, support, and everything she did for me. I had great years with you but now it is time to move on for both of us. I did not see this coming but I cannot change it either. I wish you all the best.

During the time of my PhD at Bielefeld University I was funded by a Scholarship of the CITEC, Cognitive Interaction Technology – Center of Excellence[2].

[1] `http://pfadfinder-lage.de`
[2] `http://cit-ec.de`

Contents

List of Figures

List of Tables

1. Introduction

People want to remember. They want to remember the past, special events, or just what happened some time ago. If something is very important to someone, it gets written down. The traditional way is to use paper and pen or in a more organized way a diary. A diary is a collection of memories that consists of activities and events happening in our daily lives. Along with the digital revolution of the last decade, the way of storing memories and ideas has shifted to the internet. In form of blogs, micro-blogs, and platforms like Twitter or Facebook users store, publish, and share their lives with friends and the whole world. Small portable devices made it possible to access the internet from anywhere and upload more and more information about ourselves. The level of detail of the users' content increases and the desire to maintain personal records gets stronger every day. Novel gadgets enable the measurement and collection of health related data. Overall, the main idea behind all these efforts is to record and save data about our personal lives. With all the progress in technology one major problem still exists: the amount of time to maintain and use all the systems is very high and therefore many people do not use them regularly or over a longer period of time.

Imagine a personal assistant who accompanies you all day long, wherever you go. Someone who automatically saves everything you see or experience and is able to organize, present, and share the data for you in the blink of an eye. You do not have to care to write down important information, it is just available to you. Your assistant gets to know you and is able to highlight important data automatically and is able to present the information saved in an appropriate and pleasant way. With an assistant like this the complete history of you would be available and it would easily be possible to draw conclusions from the personal history or just to revive the past.

This thesis deals with one central aspect to fulfill the above vision: the automatic detection of daily activities. Daily activities are activities people perform over the course of a day, such as working, eating, doing sports, or meeting friends. With the availability of these structuring events days can be reconstructed and patterns can be extracted. The user is able to get an overview of his or her own typical daily behavior. But also extraordinary situations stand out from the routine. The collected daily activities can be the basis for diaries, medical analyses, and self-evaluation. The idea is to support the user in journaling daily activities with the help of novel technology. While the perfect support would be a system which automatically works in the background, the goal of this work is to minimize the time which is needed to record daily activities. A key feature for such a system is unobtrusiveness. It has to embed itself in the daily life of the user and has to work efficiently with a minimum of change of the normal habits of the user. It has to be a silent, activity-recording assistant which accompanies the user in every situation.

This can only be accomplished if the system can be made aware of the situation the user is currently in. More precisely, it must be able to detect the activity or multiple activities the user performs at any given point in time. Modern technology is able to measure physical quantities with the help of electronic sensors. These are able to localize the user, measure movements and much more. As the raw data do not directly include any information about daily activities, meaningful features have to be derived which are able to represent the desired information. The feature vector from various sensors then has to be interpreted and daily activities have to be classified. Machine learning algorithms fit the given task and are able to derive daily activities from sensor feature data. In this case a multi-label algorithm has to be used because the user may perform more than one activity at the same time. As humans and their lives are highly individual, the algorithm must be able to adapt to each and every user. Finally, the system planned needs the resources to compute and store the given amount of data.

This thesis develops AMARAS (Adaptive Multi-Modal Activity Recognition Assistant for Smartphones), an application running on an ordinary smartphone collecting sensor data and storing daily activities automatically derived from calculated features. If it is not possible to determine the current situation, the system tries to support the user as good as possible by making suggestions for the current activities. The goal is to reduce the time spent by the users to keep an activity diary. The smartphone was chosen because it already is a permanent companion for many people and offers a large lineup of built-in sensors. Modern smartphones come with high computing power and large data storage options. With the phone as the targeted platform no additional device has to be carried around by the user. The already familiar interface concepts known from the smartphone operating system can be reused and no completely new procedure has to be learned. To obtain a maximum of unobtrusiveness no additional hardware is introduced, and for privacy reasons the complete computation is done directly on the smartphone. The challenge is to develop a system which is able to detect daily activities from the built-in sensors, deal with the limited resources, and finally being able to reduce the time for the user to keep a journal of daily activities. The project is realized using the Android mobile operating system for the planned demonstrator.

In a first user study sensor data were collected to get an insight into daily activity data. The analysis and evaluation of these pieces of information led to the result that it is possible to collect meaningful data and to classify activities from smartphone sensors. While the processing of the data was done offline using desktop computers, the next step was to develop and implement a system which works entirely on the device. The sensor data collection, feature extraction, and classification of daily activities are done directly on the smartphone respecting its limited resources. After achieving this goal the demonstrator application built was evaluated in a second user study. The results of this study proved that it is possible to support users in daily activity tracking by reducing the amount of work when entering information. In the given task of logging daily activities up to 52% of the needed input steps compared to a manual system could be saved. Although the application is not able to log activities fully automatically, it still reduces the work of tracking activities drastically. Above and beyond this, further improvements and additional, unobtrusive external extensions were able to enhance the classification results.

The thesis is organized as follows: chapter 2 motivates and specifies the goals of this thesis. Daily activities are defined and possible applications for the system planned are introduced. Background research and the presentation of state-of-the-art research and commercial systems is done and presented in chapter 3. Following this is a requirement analysis for the application planned as well as an introduction to currently available mobile operating systems and their respective hardware. The decisions for the following developments and studies are discussed and there are some deliberations on system limitations and resource efficiency.

With chapter 5 the investigation of daily activities and sensor data begins. An application is developed that records sensor data and combines it with user annotated activity labels in a user study. Chapter 6 analyzes and evaluates these data. After creating a suitable representation of the recorded information first classification approaches are applied. The results and experiences of the first user study are considered based on which the completely new, fully integrated demonstrator AMARAS is built to follow the initial idea of supporting users in daily activity tracking with the help of only a smartphone. The development and implementation of this application are shown in chapter 7. Finally, in the subsequent chapter 8, AMARAS is evaluated in another user study. Aside from the user experience, also the classification performance is analyzed and evaluated. The thesis closes with a summary followed by a conclusion and an outlook.

2. Research Plan for Automatic Journaling of Daily Activities

2.1. Motivation

Preserving experiences and events is a basic need for humans since they have been able to do so. The oldest known legacies are cave paintings from 40,000 years ago like the one at Bhimbetka Rock [62], India (figure 2.1). With simplest instruments the inhabitants eternalized scenes of their daily lives, such as hunting, on their cave walls with natural colors. Some of the paintings still exist, giving an exciting insight into the past. Although the original intention of these paintings is unknown, they are still important for archaeologist as sources of information. Moving on in time humans used wood or clay to paint and to write things down. 3000 B.C. the Egyptians invented papyrus [9], a predecessor of the wooden paper which is known since 200 B.C. [83] and is still the classical choice for storing information. Libraries all over the world are filled with millions of books containing knowledge, experience, and historical stories. The latest development is the digital revolution where huge amounts of information can easily be kept in digital memories, duplicated, and shared all over the world. For some areas of life paper has been completely replaced by digital information storing. Digital content is searchable within seconds and is not limited to written words only. Multimedia data like images, audio files, and movies can be attached to text to enrich the information stored.

Figure 2.1.: Cave paintings at Bhimbetka rock, photo CC BY-SA 3.0 by Wikipedia user “Conscious”

In our modern world it has become unimaginable to live without digital forms of written content. Most of the traditional paper tools have been replaced as, for example, the calendar, to-do lists, and notes. Communicating all over the world is done electronically, the biggest change in the last decade has happened to postal letters as more and more written conversation is done via e-mail or instant messaging. According to the “Statistisches Jahrbuch 2011” [Statistic Yearbook] 89% of the German internet users (75% of all citizens) were sending and receiving e-mails in 2010 [15]. We naturally use and switch seamlessly between analog and digital note-taking, e.g. writing an e-mail and using the calendar application on our personal digital

Figure 2.2.: Palm Pilot PDA, photo CC BY-SA 3.0 by Wikipedia user “BLueFiSH.as”

assistant (PDA, as seen in figure 2.2), but writing the shopping list by hand on a piece of paper. This has been made possible by small electronic devices we carry on-body in our everyday life.

One form of conserving personal events, experience, and thoughts has a long tradition in the form of keeping a diary. Typically, a diary is a blank book which is filled by the diarist. Even in the time of the digital revolution the diary is mostly done in handwritten form. Perhaps one reason for this traditional method is the fear in mind that this personal information could be copied and shared too easily and quickly when digitally accessible. There are digital counterparts to the diary like blogs or micro-blogs, but they differ in form and content as discussed later. Keeping a diary is very time intensive. Typically, the diarist relives her or his day at the end of the day and adds personal thoughts and ideas. The diary is one answer to the basic personal human need to conserve experiences of the past for the future.

For digital note-keeping high-end PDAs are available to customers in the form of smartphones and support us every day. In the last years the technology has gotten affordable to everyone and more and more people automatically received a smartphone with their mobile phone contract. Smartphones are the symbiosis of the mobile phone together with a general computing device. Equipped with a battery and small enough to fit in a pocket, it is able to store a huge amount of written and multimedia data. Additionally, a variety of sensors are built-in, able to collect information of the user's environment like location, sound, and available light. So called "apps" allow companies but also individuals to extend the original functionality of the devices. The operating system manufacturers provide a complete distribution platform for apps. This makes it very easy for developers to distribute their work to every smartphone user and leads to a very direct way of reaching customers.

As the smartphone is a daily companion for more and more people, the motivation of this thesis is to utilize this device to get closer to the vision of an omnipresent personal assistant who collects every detail of our life. The smartphone is now a central device for communication, organization, and digital storage. With phone calls and text messages users keep in contact with friends, calendar and note applications help to organize the daily tasks. Visual impressions are recorded in form of photos and videos. Why not push the field of use for smartphone even further and utilize it to record daily life in form of daily activities? With a detailed outline of daily routines the life of a user can be visualized, changes become visible over time, and outstanding events can be recognized easily. The main problem of such an intense recording of daily activities is that it is very time intensive. Every change in the user's daily cycle must be recorded which is very arduous if not impossible to maintain over a longer time period if done manually. With the help of modern technology like novel sensors and mobile computing power the smartphone should be able to support the user in this task. It should be possible to reduce the time and amount of work to a reasonable degree.

2.2. Goals and Research Questions

The goal of this thesis is to develop a system which assists users in keeping a new type of digital journal consisting of recorded daily activity entries. The approach selected in this work is to detect daily activities automatically or at least semi-automatically from smartphone sensors. By minimizing the time to maintain a daily activity diary users are encouraged to keep a journal of their lives. The smartphone is the medium of choice as it provides a platform which already accompanies many people. The sensor and computing capabilities will be utilized to classify daily activities while providing a high level of unobtrusiveness as no extra or external hardware will be used.

This leads to the following main research question for this thesis:

> **Is is possible to develop a system which automatically keeps a journal of daily activities by using a smartphone as the technical platform?**

To answer this question it can be divided into five individual consecutive questions:

1. **In how far are meaningful features derivable from smartphone sensors in order to classify daily activities?**
2. **In what way do types of daily activities differ in terms of detectability for such a system utilizing internal smartphone sensors only?**
3. **What are the key requirements for an automatic system for journaling daily activities running on a smartphone?**
4. **What are suitable metrics in order to evaluate an automatic system for journaling daily activities?**
5. **What is the utility of the developed and evaluated system?**

To reach the goals first daily activity data have to be collected and evaluated. For accurate results the sensor data have to be recorded on a suitable mobile platform which has to be selected before. The evaluation of different classification approaches will show what kinds of and how well daily activities can be detected. In order to answer the question as to whether it is possible to develop a complete system for supporting users in daily activity tracking, a demonstrator has to be built. Finally, this demonstrator has to be tested in a study and the utility of it has to be evaluated.

2.3. Definition of Daily Activities

As the understanding of personal daily activities is highly individual, they have to be defined for this thesis. For some areas daily activities already have been defined as in medical research and for the healthcare industry. In medical terms activities of daily life are defined as follows:

"Things that a person normally does during a day, including self-care (eating, bathing, dressing, grooming), work, homemaking, and leisure. The ability or inability to perform these activities can be used as a practical measure of ability or disability, and it may be used by insurers and HMOs[1] *as a rationale for approving or denying physical therapy or other treatments. Abbreviated ADL."* [92]

[1] Health Maintenance Organizations

As stated above, this definition is mainly used by health insurances to rate patients' nursing levels. A detailed list of abilities and activities to determine the activity level in this area was developed 1969 by Lawton and Brody [53]. This scale consist of eight categories like the "Ability to use telephone" up to "Ability to handle finance" each contain 3-5 activities like "Dials a few well-known numbers." In 1976 Katz et al. condensed this scale to six basic activities of daily living and generated an independence scale from 0 to 6 [45, 46, 47] and evolved this approach [44, 79]. There are many more scales and activity definitions available but the ones presented are the quasi-standards and used in many studies in the medical and psychological field. All these scales are clearly targeted at medical domains, elderly people, or people with disabilities or special needs. For the system planned they give a rough definition of what daily activities consist of and what the idea behind the concept of daily activities for this project could be.

The goal is to create a system which supports diary-like entries. For this, the recorded activities must fit into this setting. As humans' lives are highly individual, it is impossible to define a complete set of activities and even the degree of detail cannot be generally set. In order to circumstance the wide and large field of this research area activities as being examined in this project are defined as follows.

2.4. Definition of Daily Activities for this Project

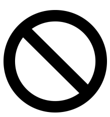
As already explained the definition of daily activities highly depends on the domain and its context. Because this thesis is all about daily activities the understanding of these has to be clear. Therefore, the term daily activities in this thesis is defined and limited by the following five statements[2]:

Must be Performable

An activity is something the user performs either actively (like walking) or passively (like going by bus). Everything else could not be measured by a system as there is no external physical impact. This limitation applies as the system must be able to detect an activity with the help of sensors.

Must be Finite

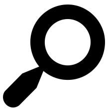
An activity is required to have a start and an end time which should not exceed one full day (24 hours). Normally an activity should last 15 minutes to 3 hours, longer activities typically are split up naturally by other short activities like breaks during work.



Cannot be Nothing

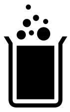
A user always performs some sort of activity, it is not possible to do "nothing." Any performable behavior of the user can be classified and assigned to an activity. Even being idle can be categorized as such. This ensures that any constellation of sensor data can be assigned to an activity.

[2]Icons by Scott Lewis (stopwatch), John Caserta (magnifying glass) from "The Noun Project" collection `http://thenounproject.com/`



Same Level of Detail

The degree of detail of activities as a whole can be chosen individually by the user but each activity should match other activities in the level of detail. So there is a general level of detail which should be honored. This important in order to keep activities comparable.



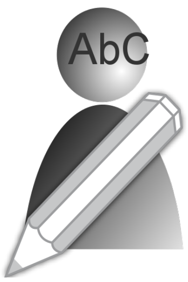
No Combinations

Each activity stands for itself and should not be combined with others to form a new activity. It is possible to perform several activities at the same time but they stand by themselves, for example, eating while watching television are two separate activities. Atomic activities prevent duplicates and are more flexible.

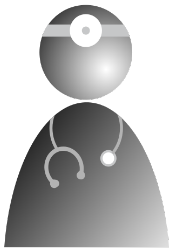
Daily activities respecting these requirements will include the presented medical daily activities but also cover potential individual peculiarities of users. It should be possible to represent all diary-like activities needed for the journal entries. Also, it ensures that it is generally possible to detect these activities with the help of some sort of sensors built-in in smartphones. Whenever referring to daily activities in this work these definitions apply.

2.5. Applications for the Planned System

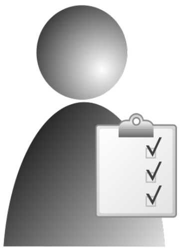
For a system like the one planned there is a large target audience. Many people could be potential users of the system and could profit from the resulting journal capabilities. In order to define the user groups, the following three application scenarios will be focused on in this thesis:



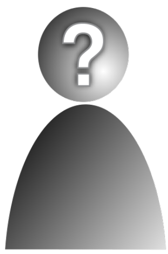
The diarist: One potential user of the system is the typical diary writer who does not want to spend too much time on the process of writing. The system supports the diarist by creating an outline of defined activities (semi-) automatically. When summing up the day these data can be used to get a rough orientation of what happened when and how long it lasted. Depending on the quality of the detected activities, only wrong or missing information has to be filled in manually. Additionally, personal notes and multimedia data can be added easily and provide extra information. A diary for special purposes, like one only for a vacation trip, can be started and maintained without much work. The result will be a diary which is very similar to the traditional writing method but will contain much more detail. The diarist does not have to rely that solely on his memory as the system provides an aid in form of saved activities. As the system can be used along everyday situations, it is easily possible to record diary data for a very long period of time. This offers long-term or even life-long diaries with a reduced amount of work. Life-loggers can use the data to associate their large media library with activities and labels automatically. This adds much more structure to their collected data. A new rising group is the quantified self community [94]. With different interests in mind these people collect not only health data to analyze behavior or changes of themselves.



Medical patients: For many mostly chronic conditions it is crucial for the doctor to know the situations the patients were in at and for a specific time. For example allergic or asthmatic patients could deduce the reasons for their bodily behavior by looking up the logged activities during problematic phases. Patients who received a long-term electrocardiography do not have to write down every single event by hand anymore because the system probably already detected the situation correctly. The generated data were objectively recorded by a computer system so the results are not influenced by unintentional or even intentional errors humans naturally make. The traditional methods always rely on the accuracy and discipline of the individual, which can be very different from patient to patient. Here, the system could provide an additional data source to compare with. In another medical field it may be a great support for patients with Alzheimer's disease. Reviving the automatically generated day plan could help remember the past or certain events. In every medical situation in which information about daily activities supports the diagnosis or helps the patient directly, this system could improve the general procedure and reduce the time of manual action. When patients used the system even before they consult the doctor, past data could help to generate a first diagnosis right at the first visit.



Change way of living: Many people would like to change their way of living. One of the biggest problems is to determine what is wrong in the current life situation and it is hard to track this down alone. An impartial system which just logs everyday life can help to get an objective overview. Based on this diagnosis a plan can be worked out to change. When fulfilling this plan the system can help again to double-check if the goals have been reached. This can be losing weight or working out more often. But also workaholics will get a representation of their work-life balance and might change this in a positive way. Overall, every time an personal objective recording of daily activities is needed, this system could provide help in monitoring daily activities. Like a personal trainer the system provides a second view on the personal day. Analytical methods based on these data can help to point out special interests and point at something that might be a problem. In a future scenario, saved daily plans of others could be compared with to each other. Users could choose and benefit from these successful plans and set them as a personal goal.



Others: Everyone who is interested in the (semi-) automatically recognition and storing of activities might be a potential user of the system. This could be a worker who is required to log events during the day for later billing or someone who needs to check if a given task has been done regularly. In a further generation of the system, detected activities could trigger events like reminding the user to do something related or start a specific application on the smartphone. Daily activities can be collected and compared against others and could be analyzed by a third party. This of course would be a dramatic privacy invasion of the users and should be applied very carefully.

2.6. Summary

Motivated by the basic need of humans to conserve memories and the past since the stone age this work aims at supporting the users in creating a personal diary-like, digital journal of daily activities. While the typical diary concentrates on extraordinary and uncommon events, this projects aims at recording recurring events and structuring them. The planned system's goal is to automatically log daily activities. Using modern technology the smartphone is a daily companion and will be the device concentrated on in this thesis. In order to preserve a maximum of unobtrusiveness only built-in sensors of the smartphone are used to determine daily activities of the users. Daily activities were specified based on medical and psychological definitions. The results are activity categories like "transportation," "meals," or "work." Also the basic conditions were set to ensure a consistent set of activities to work with. The applications give a idea of possible future user groups which should be considered while developing the system. The next step is now to take a look at already existing commercial and research systems in the area of activity logging.

3. State-of-the-Art Activity Recognition

This chapter gives an overview over current technologies and systems related to the goal of this thesis. Beginning with the latest technical development of devices, some related work of other research groups will be presented. Aside from these specialized research studies, there are also research systems with long term goals, some of which will be highlighted in this chapter. The last section focuses on commercially available systems which bring state-of-the-art technology and methods to a broad audience of consumers. The existence of these products is proof of the importance of this research area and shows already mature technologies running in productive environments.

3.1. Technical Background

3.1.1. Computing Power and Storage

According to Moore's law [76] the number of transistors in integrated circuits like CPUs doubles every two years. As seen in figure 3.1 on the left, this prediction has been proven to be true since 1971. This also applies to the chips in mobile devices, which means that they will most likely become very powerful within in the near future and will be able to process complex computations. Current smartphones feature multi-core CPUs with more than 1 GHz running full-fledged operating systems. Embedded systems are mostly customized common desktop operating systems and are no longer specialized for a single hardware platform.

A very similar development can be seen in hard disks used for data storage. While in 1997 disk space was measured in gigabyte, today devices with capacities of over one terabyte are available at a relatively low cost for everyone. Figure 3.1 on the right side show a visualization of the development over the last years. Companies like Facebook are adding more then 500 terabytes of storage to their databases every day and run disk space clusters with over 100 petabytes of storage[1]. Just like before, this development also applies to embedded devices like smartphones and their types of storage. The current standard for small memory cards is called "microSD" (for larger capacities cards that fulfill the "microSDHC" or "microSDXC," specifications are needed). These are very tiny cards sized 15.0×11.0×1.0 mm (see figure 3.2 for size comparison)

Figure 3.2.: MicroSD card (64 GB) in comparison to a 3.5 inch floppy disk (1440 KB).

[1] http://gigaom.com/data/facebook-is-collecting-your-data-500-terabytes-a-day/

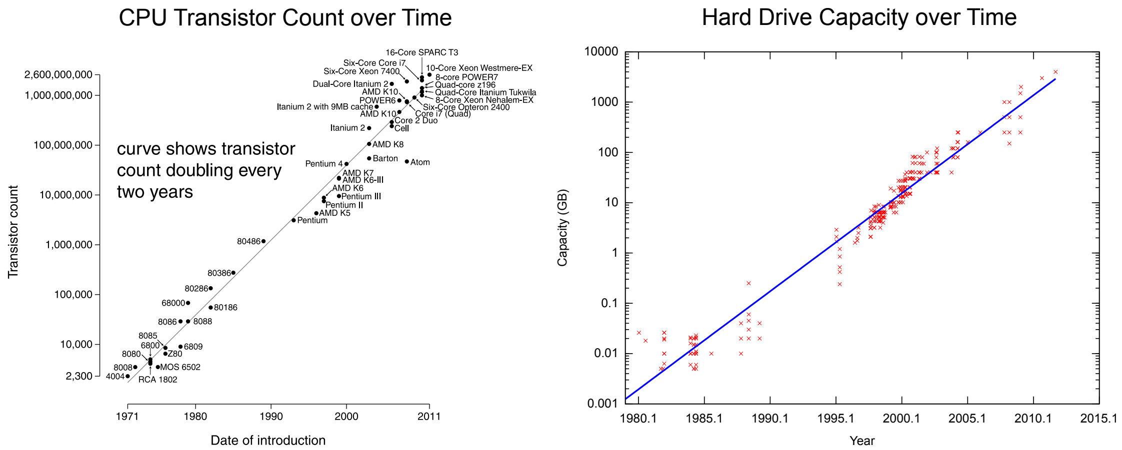


Figure 3.1.: Development of transistor count in integrated circuits and hard drive capacity over time. Data and plot from Wikipedia user Wgsimon (left, CC BY-SA 3.0) and Wikipedia user Hankwang (right, CC BY-SA 3.0)

and have a capacity of up to 64 gigabytes. This enables mobile devices to store a vast amount of data. The latest development in data storage is the so-called "cloud storage." Specialized companies like Amazon[2] sell data storage space on their server. Customers theoretically get unlimited virtual hard disk space which is accessible online and pay only for the amount of storage used on a monthly basis. This method offers a flexible and efficient way of dealing with normally limited and expensive storage space.

In summary, current devices offer very high computing power, and while data storage on these devices is limited to a few gigabytes, large hard disks are available at a low price. Current memory cards can hold about 20 hours of HD video (1280×720 pixels, 3gp codec), over 45 days of audio (128 kbps, mp3 codec) and more than 50,000 photos (2592×1552, jpg format). When there really is a need for more space on these devices, the online capabilities can be used to transfer data to external private or cloud storage. With the current technology, storage space is a limitation which can be handled easily, and the processing power of mobile devices should be fast enough to accomplish even complex tasks.

3.1.2. Small Sensors and Devices Available

In the last years some originally very complicated and expensive sensors have become very inexpensive in production and have found their way into consumer electronics. One major game changer has been the accelerometer, which has enabled the rotation of the screen orientation of a device automatically ensuring that the user is always able to read the displayed content without having to make the manual effort of turning the device itself. The sensor actually measures g-forces and can also be used for game controlling and custom measurements. As seen in figure 3.3 in orange, it is a very small chip which is able to sense three orthogonal axes. Combined with a gyroscope (figure 3.3 in yellow)

[2]Amazon S3 storage `http://aws.amazon.com/de/s3/`

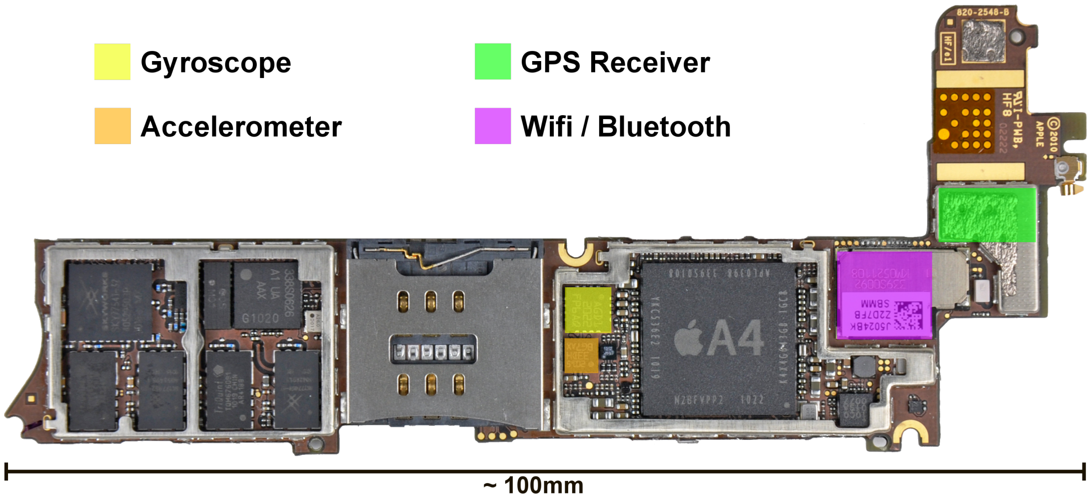


Figure 3.3.: Apple iPhone 4 logic board with sensor chip locations highlighted to illustrate the dimensions. Image courtesy of iFixit (`http://ifixit.com`)

and a magnetometer the orientation of the device can be measured with nine degrees of freedom. The low price and the very small, integrated size have made these components a standard configuration for smartphones.

The general miniaturization of components has made small devices possible. Currently the main factor for the size of a smartphone is the screen size and the battery. As seen in figure 3.3, there are three main areas on the logic board. The left side is packed with communication electronics for communicating with the mobile phone network. The largest chip is the main processor with more connectivity modules on the right. This combined chip is often called SOC (system on a chip) as it holds more than just the CPU. The sensors are located at the center of the phone. On the backside, there are mainly the memory chip and connectors. The GPS receiver (highlighted in green) and the combined WiFi and Bluetooth chip (highlighted in purple) have also decreased in size over the last years and are customarily available in smartphones. Overall, very small fully integrated devices packed with sensors are available to customers. The relatively low price through mass production is gradually allowing more and more people to afford and use smartphones.

3.2. Related Work

3.2.1. Life-Logging

A life-logger's goal is to capture continuous physiological data from his point of view. With the help of wearable recording, computing, and storing devices, this goal is achieved. The most typical way of doing this is to wear a camera facing the front capturing video continuously or photos in a short fixed time interval. Aside from the SenseCam (cf. next chapter for more information), there are several other research systems which have life-logging capabilities. The main feature of all systems is a video camera which is able to record photos and videos. As videos are very data intensive, most systems work with time or event triggered photos. The StartleCam [39] is an early video capturing system able to measure skin conductivity utilized to trigger the camera. Vemurie's dissertation project [89] is about a system called "iRemember," [90], which captures and processes audio data. This prototype is not only able to record the data but also evaluates the way in which data can be retrieved and worked with later. Timelines and tag-clouds support the users in order to find relevant data more quickly. The system has been used by the author for two years and the effectiveness of the retrieval tools has been evaluated.

Kiyoharu Aizawa's research group is about to create a complete life-logging system. The focus is on the efficient browsing and retrieval of captured data. By extracting context and meaningful information from sensors [2, 42] and various sources like the town directory [1], the users are able to access stored information. More sophisticated summary generating methods like conversation detection [3] add more meaningful information to the recorded data and enable users to retrieve information more efficiently. As the feature extraction from sensor data seems to be the key to combine life-log data with context, Aiden et al. utilized multimodal features for the identification of important events in life-logging data [26]. Above and beyond that, face-detection and the concept of novelty have been added for more detailed context information [25].

Aside from the typical life-logging approach, the data recording methods can be used to support people with, for example, memory impairment. Based on previous work [55], Lee and Dey utilized life-logging to help patients with episodic memory impairment remember the past better [56]. Because of the collection of all these intimate data, privacy is a huge concern and must be respected. The master's thesis by Chaudhari [19] addresses exactly this problem. While all the projects presented so far were driven by research facilities, there is also a community of technophile individuals who are interested in collecting data for private use. This community is very similar to the quantified self people, who will be presented in the next section.

Overall, the tools and devices to capture daily life data are available and ready-to-use. The main research area here is how to deal with the enormous amount of data. Video, audio, and other sensor data has to be connected to context information in order to make the collected data accessible for the user. Without this refinement, it is just a gallery of moments which is barely manageable and useable for humans.

3.2.2. Quantified Self Community

The aim of quantified self people is to record data about themselves and try to get to know their body better by interpreting these data on their own. They use all kinds of tools to achieve this goal. On the one hand, this can be pencil and paper writing down their body weight every morning or, on the other hand, some kind of tracking apps or specialized gadgets like the Fitbit device, which will be presented and discussed later. The rise of connected sensor devices has improved the quality and quantity of data that can be measured by a layperson and, therefore, pushed the community forward in terms of accuracy and amount of data that can be collected. The result can be personal findings about specific individual circumstances which have led to certain effects, but also the ability to compare data with other quantified self followers. The aim of the community is to have the personal advantage of knowledge by storing and analyzing as much data as possible about their daily lives. For this reason, the system planned for this thesis will be an additional tool which enables users to connect data with activity labels, on the one hand, and the raw sensor data which have been collected and can then be used for further analysis, on the other hand.

The quantified self movement was originally initiated by the *Wired*[3] authors Gary Wolf and Kevin Kelly in 2007. The main information source of the community is the homepage `http://quantifiedself.com`, which lists meetup groups all over the world. Gary Wolf gave a TED[4] talk about the idea of quantified self in 2010 [94]. In Germany, there are only a few groups loosely organizing mainly using a Facebook group and the homepage `http://qsdeutschland.de`.

3.2.3. Localization, Location, and Transportation

The knowledge of the current location of a user is an important key for detecting activities. Many activities are directly connected to places. Many places and buildings were especially built for specific activities like supermarkets, theaters, or gyms. When examining the locations over time, the speed between spots can be calculated and this hints at the type of transportation a person uses. This can be staying at one place, going on foot, or using faster methods such as taking the car, bus, or train. The standard built-in method in consumer mobile devices for localization uses the satellite based Global Positioning System (GPS). Still, there are drawbacks because of the relatively high power consumption and the long interval for the first position results (time to first fix). Also, GPS does not work well inside buildings and cities with skyscrapers. Yet, the position is crucial when trying to get more information about the current activity. There are several research projects which demonstrate alternate localization approaches and how to utilize this information to determine daily routines and situations.

Localization by using the *Global System for Mobile Communications* (mobile phone network, GSM) and the *Universal Mobile Telecommunications System* (UMTS, 3G network) cell tower information or *IEEE 802.11* standard (WiFi) enabled access points is commonly used by and integrated in smartphones. Aside from the vendor-provided services like the one by Google or Apple, there are also third-party companies: Skyhook[5] sells this localization approach as a service. In 2005, early works by researchers investigated this new method for user localization [52, 87]. This new kind of triangulation based localization works very well in metropolitan areas as there are many access points and cell towers available [20]. In figure 3.4 there is an incomplete map of estimated cell tower locations in Munich, Germany, of one provider allowing to get an idea of how many cell towers are available in one area; it is taken into account that there are three more major providers available in the country. The idea of radio frequency-based localization is not new [5] and even accurate indoor localization is possible by using GSM/UMTS cell towers [68] or available WiFi access points [37, 51]. Recently, Google added indoor localization to their “Maps”-application enabling users to locate themselves and plan routes even inside public buildings[6]. Specialized Bluetooth beacons placed in rooms are able to provide high accuracy localization [32]; this has already been tested for life-logging purposes [64]. An-

[3] `http://www.wired.com`
[4] `http://www.ted.com`
[5] `http://skyhookwireless.com`
[6] `http://support.google.com/gmm/bin/answer.py?hl=en&answer=1685872`

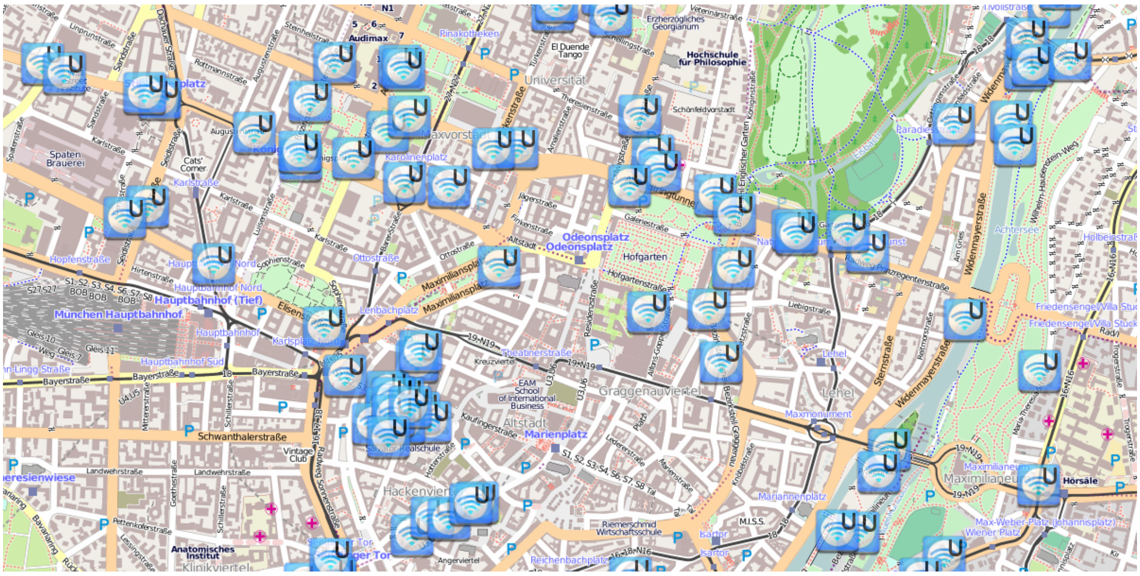


Figure 3.4.: Incomplete map of estimated cell tower locations in Munich for one provider. Data have been acquired with the help of a community using an application which connects the position to currently connected cell tower IDs. Data: `http://cellhunter.omoco.de`, CC BY-SA 3.0 / Openstreetmap `http://openstreetmap.org` CC BY-SA 2.0

other relatively new approach is to compute ambient fingerprints from various sensors of a smartphone for location recognition [4].

GSM-/UMTS- and WiFi based localization is a well-proven method to locate a smartphone user and can be even more accurate than GPS systems. In contrast to GPS, the power consumption is lower but the accuracy and reliability is highly depending on the area the user is in as the methods are based on existing beacons and the quality of information about these beacons. In the end, it is a nice, fast addition to GPS positioning. One drawback here is the need for a local beacon database on the device or a working internet connection to query a service provider. For high accuracy, indoor localization always requires prior mapping of the buildings and, therefore, is only applicable in some specialized situations.

After retrieving the current location of a person, the next step is to connect meaningful information to it. In order to investigate this field, a large set of data has been collected. Nathan Eagle and Alex Pentland conducted a nine-month study in which 100 users equipped with Nokia 6600 smartphones collected cell tower IDs for location approximation, nearby Bluetooth devices, and phone usage data [27]. A subsequent study used this data to evaluate the structure in daily routines [28]. As the collected information is publicly available[7], several other studies followed up. With their paper "What did you do today?", Farrahi and Gatica-Perez developed a method which automatically discovers characteristic routines for individuals [29, 30] and tried to predict daily life patterns [31]. Lin Liao et al. [58] added high-level context information to detect significant places of persons using GPS location data. In order to create training data for such systems, a small device called "LifeTag" was developed by Rekimoto et al. [73] and is based on their

[7] `http://reality.media.mit.edu/dataset.php`

“PlaceEngine” [72] device. Worn on a key chain it scans for wireless networks in a fixed time interval and stores them on flash memory. A WiFi-database later translates these data into location information. Additionally, it is equipped with a button which is used to bookmark places of interest. This is a very simple and elegant way of storing and connecting locations with additional information.

Transportation is not only limited to being transported, such as going by bus or driving the car. Transportation is the way of getting from one place to another. This can be done either actively by walking or passively by taking the bus. Zheng et al. evaluated several classification approaches based on raw GPS location data [98]. Aside from the location information, the accelerometer is a good sensor for detecting types of transportation. While Nham et al. were using the accelerometer data only [66], Reddy et al. proposed a combined approach with additional GPS data from mobile phones [71]. High-level properties of transportation and mobility can also be detected by using GSM cell-tower information only [80].

With the help of various localization approaches and the GPS system, the current location of a user can be determined. As seen in the studies presented, the location is a strong indicator for situations, activities, and routines. Knowing the type of transportation is another very good clue for the current place the user is in or going to be in.

3.2.4. Physical Activity Tracking from Accelerometer Data

In general, there are three different approaches which have to be distinguished in physical activity recognition from accelerometer data. First, there are some studies which place many accelerometers all over the user’s body to get as exact a result as possible[17]. This is a very promising way but absolutely unacceptable in everyday situations. Second, with the upcoming of small wearable mobile devices there are several designs using a single two- or three-axis accelerometer which is able to collect and transmit data to an external computing device. This enables easy and unobtrusive handling of the device; also high quality sensors can be used. Thirdly, there is the recent generation of cell phones and smartphones with which companies discovered the accelerometer as a standard sensor to set the screen orientation and also as a device to control games. These inexpensive consumer electronics are used in current research enabling a completely unobtrusive approach because the user already owns the smartphone and the sensor data can be collected along with the daily usage. On the other hand, this approach has to make do with low precision sensors and cannot rely on a fixed position of the sensor.

Figure 3.5.: The activity recognition system with wired accelerometers as used in the publication of Bussmann et al. [17]

In order to measure the activity of ambulatory patients, Bussmann et al. developed a system which is able to distinguish between static (standing, sitting, and lying) and dynamic (walking, ascending / descending stairs, cycling) activities. For this reason, in 1996 uniaxial accelerometers were placed at different parts of the body [88] as seen in figure 3.5. This study was refined with newer hardware and the placement of the sensors on the extremities [17]. A similar approach was tested by Makikawa et al. [60] who placed the sensors on the joints. They arrived at the conclusion that their method is better-suited for detailed posture change and behavior detection than single accelerometer systems. Aiming at the classification of 20 different activities, Bao and Intille equipped their participants with five self-sufficient bi-axial accelerometers [6]. Placed on strategic places all over the body, they achieved very good results. Several other studies confirmed that it is possible to classify physical activities with the help of accelerometers worn on the body.

While it might be suitable for short term or outpatient medical observations, it is not a practical approach in daily activity recognition to use multiple accelerometers. Even in outpatient environments the use of only one small device will save time for the patients and enhance the comfort of using the system. Concentrating on the very basic activities such as resting, walking, and sit-to-stand transitions, Mathie et al. developed a very small, light but efficient device containing one two-axis accelerometer [61]. Advanced and enhanced methods use newly available three-axis accelerometers to cover more activities [70]. As nowadays a three-axis accelerometer is available to consumers built-in in smartphones, the previous studies have been ported to these small new self-sufficient computing devices. Sprager and Zazula were able to distinguish between speeds of walking with a cell phone attached to the person's hip [81]. Based on the traditional physical activities from the earlier studies presented, Iso and Yamazaki attached an external accelerometer to a phone [43] and later Brezmes et al. proved that this method also works with consumer phones [13]. The latest addition in this field is an implementation of these well-proven methods on Android smartphones by Kwapisz et al. [50].

For physical activities, especially for gait and movement analysis, the accelerometer-based approach is very promising. Even if a detailed analysis is not possible with only one accelerometer, it is still a good method to detect activities and contribute valuable information about the current situation.

3.2.5. Medical Use of Diary Systems

Aside from outpatient monitoring of physical activities, there are other potential fields of use in medicine. The study by Heiberg et al. has shown that when recording the daily health status, the traditional pen-and-paper method is comparable to a digital version [40]. Actually 82.9% of the participants preferred the digital method. Russel et al. presented a diary method which provides a qualitatively and quantitatively more precise method of diagnosing migraines [75]. There are applications for private use for smartphones available today which help users in their self-diagnosis. One of these applications is shown in figure 3.6. It enables the detailed recording of headaches and offers a statistical evaluation of occurrences[8]. Overall, there are many approaches to replace the traditional logging and documenting method with digital counterparts. Especially the subsequent evaluation of the data gets much easier as it is already digitally available. It will be only a matter of time

[8] "Headache Diary Pro" by appcellent GmbH, `http://itunes.apple.com/de/app/kopfschmerztagebuch-pro/id402258244`

Figure 3.6.: Screenshots of the app "Headache Diary Pro" for iPhone. It offers detailed recording of headaches which supports the diagnosis of migraines. Input of a headache on the left, history in the center image, and statistical analysis on the right.

until the current problems are eliminated and the pen-and-paper method will disappear from the daily work flow in medical use.

3.3. Research Systems

3.3.1. Memex Vision by Vannevar Bush

The Memory Extender (*memex*) is a hypothetical prototype of a device which was envisioned by Vannevar Bush in his 1945 article "As We May Think" [16]. Aside from the memex vision, the article covers the current situation of researchers at that time and the technical advances that had been achieved in the years before. At the time of writing, the photography on microfilm had already been improved and used heavily. This also is the basis of Bush' idea. Bush proposed a desk-like device the user operates by entering and retrieving information about his life and general facts from encyclopedias. Information is available on preprocessed microfilms and the user is able to browse and search the content. Personal notes and comments can be added and content belonging together is presented automatically. The desk itself consists of screens, a keyboard, and several buttons and levers, but also the idea of remote operation is addressed by him.

Even at this early stage Bush envisioned the fact that a vast amount of data comes along with the problem of sorting and organizing it. The fundamental feature of the memex device was that matching information is retrieved immediately and automatically when requested. He stated out that "[T]he Process of tying two items together is the important thing"(§7 of [16]). Bush proposed so-called "trails" of information which can be created by the user that bind two or more objects together permanently. Trails can be named and tagged with keywords and reflect interests of the user. Trails are permanently stored, and even the idea of sharing trails with other people is mentioned in the article. Every time the user accesses an object connected to a trail, it becomes available and offers further information stored. This concept is one of the first mentions of the hyperlink concept now commonly used by HTML technology in the web [67].

Vannevar Bush was the first researcher to get a lot of publicity with his article about a *memex* vision, but the idea itself is a little bit older as described by Bucklands in 1992 [14]. First thoughts about this vision were published in the 1920s by Emanuel Goldberg of Zeiss Ikon in Dresden, Germany. His patent from 1927 was the first appearance of the idea to organize and manage photos automatically. In 1932, his work *The Retrieval Problem in Photography* [36] was published. This means that the general idea to create such a *memex* device is not new but Bush was the first one to write it down in detail and get the attention of a broader community. In any case, his visionary work has inspired many projects which aim to achieve his stated goals such as the MyLifeBits [33] project by Microsoft, which will be presented next.

3.3.2. MyLifeBits by Microsoft

"MylifeBits is a lifetime store of everything. It is the fulfillment of Vannevar Bush's 1945 Memex vision including full-text search, text & audio annotations, and hyperlinks."[9]

These are the introductory words on the official project homepage describing MylifeBits in short and linking it directly to the presented work of Vannevar Bush. The project is divided into two parts: First, there was an experiment in which researcher Gordon Bell collected a huge amount of data from all kinds of sources starting in 1998 [8]. The second part is the attempt of getting an overview and the control over this enormous amount of data by hyperlinks, annotations, clustering, and similar techniques. It is about creating software and methods to gain benefit from the collected data for the user [33]. Just as the original *memex* article before, this big project got much attention by the press and is relatively well-known worldwide.

Figure 3.7.: Gordon Bell wearing a Microsoft SenseCam, photo by Queensland University of Technology

In order to fulfill the first part, Gordon Bell (cf. photo 3.7), born 1934, himself took part in his experiment by collecting all kinds of data in and of his life. He tried to digitalize all media he ever consumed or had produced. This means that every single book, musical piece, and video ever consumed in his past as well as other personal data was saved in his digital library [7]. In a next step he started to collect information about his everyday life by collecting any kind of conversation with others as well as browser history and even the websites browsed. He gradually developed tools to get this information in a more and more automatized way. His personal goal is to eliminate all non-digital media in his life. From 2001 to 2007 he collected about 300,000 items in his database, which uses about 150 GB of disk space. Over 100,000 websites and e-mails, 15,000 text files, and 2,000 presentations have been stored so far [8].

[9] `http://research.microsoft.com/en-us/projects/mylifebits/`

The second part and sister project of the MyLifeBits project's target is to master all the collected information. In a first step the aim is the digitizing process, so every spoken conversation, every text or letter has to be available in digital form. In a next step, the team developed a fast, fully searchable storage for this huge amount of information which can hold any type of data [34]. Several tools have been developed for feature extraction from images, videos, and audio files; in addition, everything is operated with a graphical user interface. Location information can be connected to recorded images and videos, conversations with individuals, and archived websites with their corresponding notes can be accessed directly from the browser while surfing the web. Overall, the original ideas of Bush's trails are implemented and refined step by step.

3.3.3. SenseCam by Microsoft

One device developed by Microsoft to fill up this lifetime database with information is the SenseCam as shown in figure 3.8. It is a small self-contained device worn on a strap around the neck. This device is able to take photos autonomously with a fish-eye lens in order to maximize the field of view. It was invented by Lyndsay Williams in 2003 while he was working for Microsoft Research[10]. The camera takes pictures at VGA (640×480 pixels) resolution, for storage a SD card slot is available. Along with the pictures, the timestamp, accelerometer data, and location estimation via GPS are recorded. In addition to the accelerometer, a light-intensity, a light color, and a passive infrared (body heat) sensor is used to automatically trigger the camera when the user is in a potentially interesting situation [41]. Originally, only developed to be a research device for Microsoft internally, the SenseCam is now publicly available and distributed by the Vicon company as the Vicon Revue 3MP[11]. The technical specifications have since been improved and the camera now has a 3 megapixel sensor, a magnetometer as well as a temperature sensor.

Figure 3.8.: Microsoft SenseCam, photo CC BY-SA 2.0 Fr by Wikipedia user "Rama"

In 2007, Sellen et al. showed that the regularly taken images from the SenseCam can support humans in remembering the past or trigger memories by reviewing a set of pictures [78]. Inspired by the great amount of picture data, Lee et al. developed a system to manage the pictures with a novel interactive photo browser [54]. Overall, the SenseCam is used in many research projects[12] as it is a simple, self-containing, relatively inexpensive (£ 299) device. The affordable price of the SenseCam also attracts personal life-loggers all over the world. Example images taken with the SenseCam can be seen in figure 3.9, showing six prototypical images in a day of a lifelogger wearing it.

[10] http://research.microsoft.com/en-us/um/cambridge/projects/sensecam/questions.htm
[11] http://viconrevue.com/product.html
[12] http://research.microsoft.com/en-us/um/cambridge/projects/sensecam/publications.htm

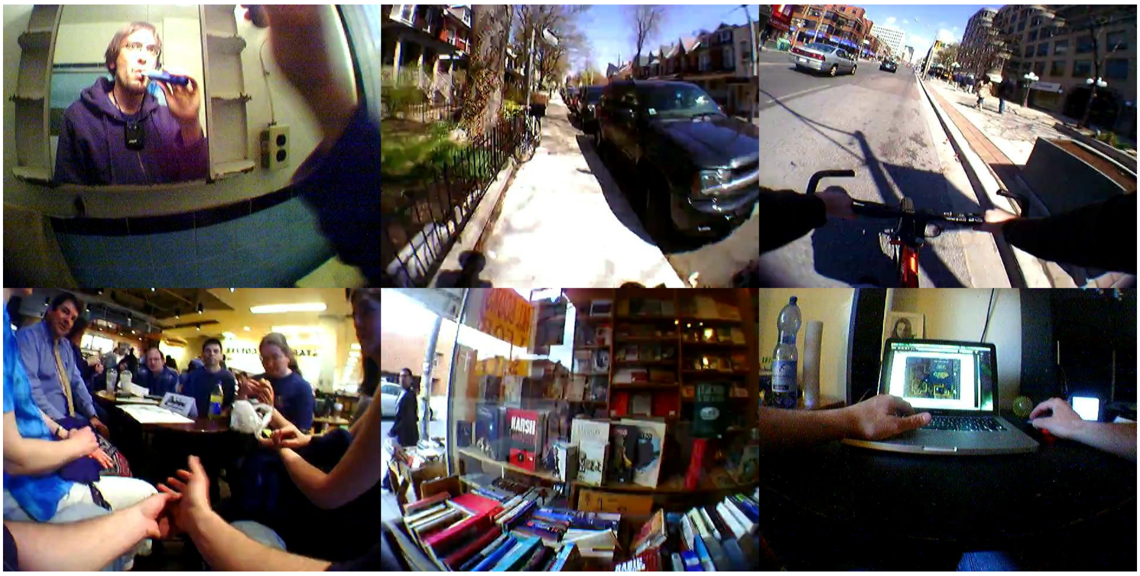


Figure 3.9.: Pictures taken over a day with the Microsoft SenseCam. From top-left to bottom-right: Morning hygiene, walking outside, riding the bicycle, meeting with friends, visiting the bookstore, playing games at home. 2009 by Janusz Leidgens, available online: `http://vimeo.com/4540866`

3.4. Commercial Systems

3.4.1. Classic Diary and Digital Counterparts

"A diary is a record (originally in handwritten format) with discrete entries ordered by date reporting on what has happened over the course of a day or other period. A personal diary may include a person's experiences, and/or thoughts or feelings, including comment on current events outside the writer's direct experience." [Wikipedia[13]]

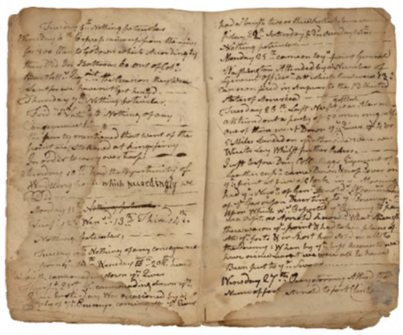


Figure 3.10.: Historical handwritten diary ("Samuel Leavitt's Journal to Westpoint")

First occurrences of diaries are dated back to the time before Christ, for example the GADEx texts [82] of Babylonia reporting astronomical events. In general, diary entries are very personal and reflect the experiences and thoughts of the author. The entries are written regularly but there is no general style of writing. Depending on the diarist, it may be a very simple, artless style up to a linguistic masterpiece. Entries can be interdepend but mostly these are recent fragments of important events in the life of the author. The content does not even have to be in chronological order depending on how well the writer prepared the entries. Often a diary is used for self-reflection and to reprocess the experiences of the day. Historical diaries give an unfiltered insight into the life of a person and the times he lived in and are for this reason very

[13]`http://en.wikipedia.org/w/index.php?title=Diary&oldid=490765509`

important findings for archeologists and historians. Exceptional circumstances like wars, social or political situations cause many people to start and maintain a diary.

Since the late 1990s, modern web publishing software opened the web for non-technical users. From this moment on it has been very easy to share texts on the internet, which in turn has led to a quickly rising and still growing number of so called "blogs." The word derives from the term weblog, which describes a kind of diary publicly available online [65]. It is possible to host a blog on self-owned webspace by using software like WordPress[14] or, as done by the majority of bloggers, to use a blog-publishing service like Blogger[15]. The software services can be mostly used at no cost at all and without any prior knowledge and can be personalized by the owner. The major drawback compared to self hosting is that the content is given to the service provider and, depending on the terms and conditions, some rights of the contents are reserved. Yet the ease of use opens web publishing and blogging to everyone, which has made this services very popular.

Figure 3.11.: Tweets from @BarackObama (12/07/12), he used Twitter heavily in the election campaign 2008

A blog typically consists of reverse chronological posts written by an individual or by a small group. There are also dedicated blogs by companies where employees post in a larger group. The written content can be enriched by media like images or videos and sometimes only concentrates on those. Podcasts are audio-only blogs where the diarist records a file which can be subscribed to and downloaded by the visitors. In contrast to the original diary, the online counterparts differ slightly in content. While the classic diary primarily contains very personal experiences and thoughts, a blog is more about things other people are likely to read and interested in. On the whole, the posts are mostly not especially intimate as blogs are generally available to the public. Most blogs are specialized on a topic like politics, shopping, lifestyle or just about daily life or holidays. As blogs are not centrally organized, many of them offer the possibility of subscribing to updates via RSS feeds. Using a feed reader, all blogs the reader is interested in can be accumulated. In 2006, Twitter[16] started a novel kind of blogging service where each post only allows 140 characters [49]. This form of publishing is called micro-blogging [97]. The character limit leads to very concentrated "tweets" with news, short updates about the user's life, or simply short statements (see screenshot 3.11 for an example). Users can follow others to receive their updates automatically, reply to tweets, and set topics with "hashtags." On Twitter, news nearly instantly spreads all over the world and this is nowadays the fastest way to publish a message and reach many people, and given that it is important enough, the own followers "retweet" it to their followers, generating a snowball effect.

In sum it can be said that the new kind of online diary-keeping is only partially related to

[14] http://wordpress.org
[15] http://blogger.com
[16] http://twitter.com

the original diary. The entries and also the addressees differ as blogs are generally public and are mostly used to share experiences while the diary is not designed to be read by other people in the first place. There is some overlap in content, and also the form of writing is very similar. The new digital way offers more and other options and therefore the classic diary still exists in its own right.

3.4.2. Fitbit Ultra

Fitbit is a company start-up located in San Francisco specialized in health and sports tracking. They provide a website[17] where users can keep track of their activities, weight, food and other health related values. In the freemium version they offer some basic analyses and the option of defining personal goals. This is, for example, for reaching a specific weight or a minimum count of steps per day. The user is able to share the results with the community and earns badges and awards for goals achieved. Like other social networks, friends and connections to other users can be established. The idea behind this is to cheer each other for better results. A premium account for 49.99$/year[18] offers more sophisticated methods, reports, benchmarking and ranking tools. Also a so-called "digital trainer" is available which personalizes the fitness plan individually for the users and is supposed to encourage them like a normal fitness coach to reach their goals.

Figure 3.12.: Fitbit Ultra device, photo CC BY-SA 3.0 by Wikipedia user "Ashstar01"

The whole company and website are built around their product called the "Fitbit Tracker" as seen in figure 3.12. This is a small peg-sized device which can be bought directly from the Fitbit store for 99.95$. The main component is a three-axis accelerometer and an OLED-display. Using these components, the device keeps track of steps, distance walked and estimates the calories burned for the current day. By using the only button on the device the data are displayed. The latest generation device called "Fitbit Ultra" is also able to show the number of floors climbed, features a digital clock, and can be used as a stopwatch. As a bonus the device can be used while sleeping to get a rough overview of sleep cycles and nightly activity. In order to collect this data, the device must be worn on the wrist while sleeping. When the device gets near its base station, it automatically connects and uploads the data stored to the website every 15 minutes. This is done by the extremely power efficient ANT[19] hardware and wireless protocol. Under normal conditions the battery lasts for a whole week and can be charged by sticking the Fitbit Ultra on the base station.

In order to gain firsthand experience on how the platform and the device work, I personally tested it for about one month. These are the subjective results: The operation and handling of the Fitbit Ultra is very simple. Its only button is used to cycle the views

[17] http://fitbit.com
[18] http://fitbit.com/premium/about
[19] http://thisisant.com

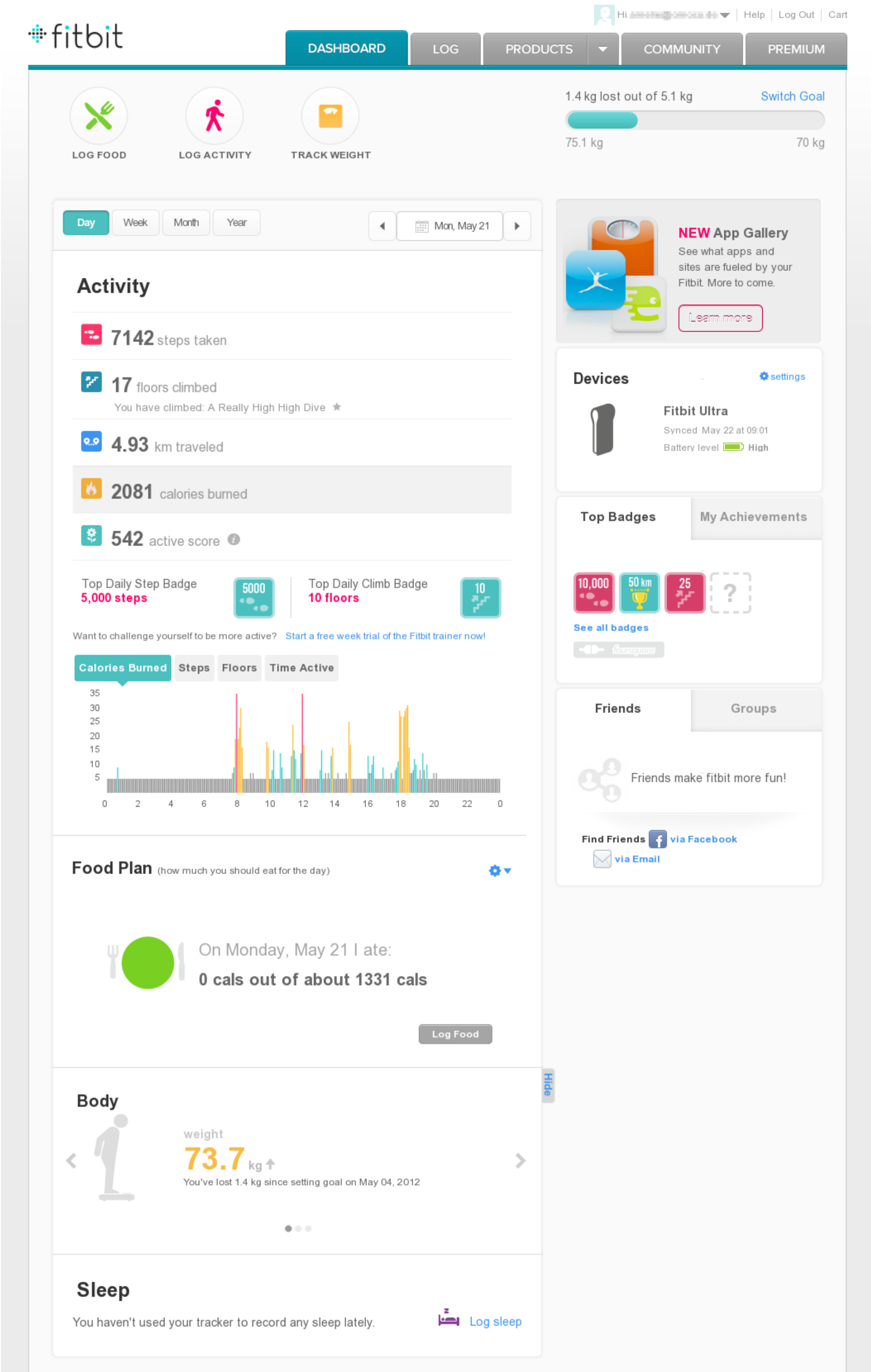


Figure 3.13.: Fitbit homepage, dashboard view. Here, the summary of all information is visible. At the top the buttons to log food, activities and weight can be seen. "Activity" displays and visualizes the information gathered by the Fitbit Ultra device. The food plan calculates the calories left for the day to reach the set goal and the weight information is available. Optionally, the tracker device can be used to collect sleep information when worn over night.

on the display and a long press starts and stops a timer which can be used for labeling time slots later. When picking up the device, a “motivating message” is displayed and it starts to count the user’s steps. Aside from the numbers, a growing flower in the display represents the activeness for the current day. The device is very small and can be worn on the belt or directly attaches as a clip to the pants. Altogether it is a very well designed piece of hardware fulfilling the desired task perfectly. Every time the tracker is near the base station it synchronizes saved data automatically. The station must be plugged into the USB port and a small daemon program has to be running on the Windows or MacOS computer. The battery life is very good and it is no problem to keep the device charged for over a week.

After logging onto the website (cf. screenshot in figure 3.13), the dashboard is displayed with the collected information. The summary displays recent activity values, the food plan and weight data. By setting a goal, these data get customized so the user always sees how many calories are left to eat for the day. The number of steps, floors, and the way traveled are automatically collected by the tracker. With the help of personal information like sex, age, and body height, the burned calories for the day are estimated. Logging activities and weight is done easily by entering the data into the web form. It gets a little more complicated when entering food. Meals and single items of food can be selected from a community-generated database and if not available, it can be added to the database. Therefore, the user has to trust the nutritional data entered by other users; there is no supervised information available. This procedure can be very time consuming but is necessary in order to get proper calorie suggestions.

Figure 3.14.: Fitbit Aria body scale, photo CC BY-SA 2.0 by Flickr user “justgrimes”

Overall, the platform can help users keep track of their daily activities and also help to reach goals like weight loss or engaging in more sports. Badges and awards are used as motivation and the results can be shared with friends, which is supposed to be additional motivation. Wearing and using the Fitbit Ultra is not the main problem but the manual part is, because, for example, entering food data gets more and more exerting each day. Also, the tracker is an additional device the users have to take with them and take care of every day; this can be tedious. Therefore, it is hard to maintain this routine for a longer time. The company just released a new and interesting device called the “FitBit Aria Smart Scale”[20] as seen in figure 3.14. This WiFi enabled body scale automatically transmits the weight to the website and automates another part of the tracking process. Now, only the entering of food has to become easier and Fitbit could really provide everyday useable health tracking. The social aspect is very interesting because users get motivation from friends and can compare each other to reach shared goals. In summary the combination of the presented devices could really help people to live a healthier life without spending much money on a personal trainer.

[20] `http://fitbit.com/product/aria`

3.4.3. Facebook (Timeline)

Figure 3.15.: Facebook Timeline view, screenshot CC BY-ND 2.0 US by Flickr user "Jayabharat"

Until the end of December 2011, Facebook profiles only showed recent events and photos of their users. This changed with the introduction of the so-called "Timeline." The new look of the profile page enables the users to order events, photos, and videos of their entire life chronologically (see screenshot 3.15). This visually refined look differs from the originally personal site by highlighting events of the users and also displaying notifications from connected applications. A cover photo dominates the top of the page, below is the user's profile information and some friends' activities. Then the important things follow which can be chosen and added by the user. Normal posts on Facebook have to be marked manually to be featured on the Timeline. On the right hand side there is a navigation bar where the viewer can quickly scroll back through the already logged months and years.

The interesting part of this new feature is that the users are encouraged to select events from their daily lives, save them for later and show them to their friends. This leads to a very vivid kind of diary minted by images and videos. It is meant to be a visual tour through an individual's life. It is generated automatically from the usual posts of a user and with the integration of the installed Facebook applications; the information is even posted automatically. Facebook introduced the term of "frictionless sharing"[21] for this procedure. The user does not have to take action to publish or post information about himself. Each application asks once for the permission to add things to the Timeline and can act autonomously in the future. This can be results of games played but also information taken from hardware like FitBit. Although Mark Zuckerberg mentioned this term at the F8 conference[22] as a new invention, it was defined earlier by Weller [93].

3.5. Summary

The latest development in consumer electronics has led to a distribution of small, integrated devices to millions of people. In form of smartphones they are carried around with their owners every day and are available on-body nearly 24 hours a day. Featuring a fast CPU and a lot of memory, each one is more like a small computer than a cell phone. Storage capacity has reached dimensions in which myriads of information can be saved and extended by cloud services; when needed, a whole series of sensors from accelerometers over compasses to gyroscopes is packed into the phones. Usually there is a GPS receiver available for localization. Otherwise or additionally the information from connected cell

[21] http://en.wikipedia.org/w/index.php?title=Frictionless_sharing&oldid=486992624
[22] https://f8.facebook.com

towers and available WiFi access points can be used. Overall, there is a sensor box available which can be afforded by consumers and which fulfills all requirements needed for activity recognition.

The availability of these sensors and the desire of humans to know each other has formed communities which are all about measuring every bit of information about their body and life. Life-loggers and the quantified self community are a phenomenon of the last years. People record their daily lives and share their knowledge with like-minded people. On the one hand, the goal is just to record any kind of data available but, on the other hand, the idea is to consolidate occurrences of body behavior with past data. Following the *memex* vision, a person's entire personal life is accessible at the computer and can be archived, searched, and shared. With the MyLifeBits project, Microsoft is pursuing this goal and has invested a lot of money into this research. Hardware like the Microsoft SenseCam was developed and enables the researchers to collect data more easily and continuously than before. First long-term results are available and further worked on. As the mere collection of pictures, videos, sounds and environment information is already possible, the current research focus is on how these data can be organized.

Because the current location of a user and context information of this location is very important in activity detection and tracking, there are several approaches of researchers that determine this and that have to be considered. Localization itself is only the first step and can be done in many ways which differ in energy costs, accuracy and their capability to work correctly in the current situation. In general, localization methods are available as built-ins in smartphones and work fine. Going one step further to activity recognition, the transportation mode can be derived from locations over time. To figure out activities at one coarse spot but with significant body movement, the accelerometer plays an important role. Researchers started in the nineties with equip-able accelerometers all over the body but also showed that information can be concluded from the built-in accelerometers in smartphones.

In some areas these technologies have already arrived in our daily lives. Sports- and movement-tracking consumer devices like the FitBit Ultra use accelerometers to help people become more active. With a detailed analysis of the daily behavior, it acts like a personal coach and helps to reach personal goals in order to get fit and to live a healthier life. The diary aspect is used by people every day in form of Twitter and Facebook. These companies are specialized in collecting information about daily life and present it using the Facebook Timeline to the rest of the world. With over 900 million users[23], Facebook proves that people like to write down their everyday experiences and also like to share it with their friends.

All these already very successful projects must now be combined to reach the goal which is a system that supports the creation of a digital activity diary. The combination of the specialized projects presented in activity recognition should cover most of the information needed in order to detect activities of daily life. The general approach planned might be not as accurate as the single counterpart but, overall, the result shown are very promising. The main objective is to create a system which fits unobtrusively and seamlessly into our daily lives, and for that purpose the smartphone as an integrated device is just perfect. As our natural daily companion all the built-in sensors are able to collect the data needed

[23] "955 million monthly active users at the end of June 2012." (`https://newsroom.fb.com/content/default.aspx?NewsAreaId=22`, 23-Sep-2012)

and the processing power is also able to handle the information. Activity predictions can be made and support the user in creating his personal digital diary of daily activities. The next step is now to get an overview of the smartphone market, choose the right device and the best fitting tools. Then exemplary data are required to test methods or recording, analyzing, and evaluating the information recorded.

4. Mobile Platforms and Technology Overview

The development of an activity recognition system which uses a smartphone requires a close look into currently available platforms. With the decision to use a smartphone, the first questions after a requirement analysis is to decide which mobile operating system best fits the needs. This chapter begins with an overview of state-of-the-art smartphone platforms and their features and limitations in software and hardware. The next step is to get and present an overview and select the available hardware on the market. The phone chosen will be introduced and there will be a detailed description of the built-in hardware and sensors. Decisions on the software development toolkit will be argued and also thoughts about the design of the user interface will be shared. The second to last as well as the last part will be the breakdown of possible pitfalls and limitations of the selected hardware and software and the question as to how to deal with the limited resources available such as battery life. Finally, some ideas for extending the system by external sensors for better and more precise activity recognition will be presented.

4.1. Requirement Analysis

In order to reach all the targeted goals of the system, the limitations given by the smartphone have to be kept minimal. This means that the mobile operating system that is chosen has to, on the one hand, allow for access to the hardware and sensors as directly as possible and, on the other hand, has to offer prototyping capabilities. It is very important when creating a research system to be able to quickly and efficiently develop software and to test several methods without losing too much time. In addition, direct access to the hardware layer has to be enabled in order to get the last bit of information out of the sensors of the selected platform. This guarantees pure data which have not been obfuscated and simplified by a software layer. This is often the case when manufacturers try to meet the needs of the casual application programmer who does not need the full range of information and degree of detail of the built-in hardware. The operating system has to provide a deployment process that offers an easy, fast, and autonomous distribution of applications on testing devices.

The software will not be able to generate good results if it runs on bad or slow hardware. A handset with sufficient or even over-designed hardware is needed for best results. Only high-end products will fit these needs and ensure that the hardware is not the bottleneck of the planned system or at least push the limits as far as possible. The implementation of the system should be mainly a software engineering process and not a hardware project as well. So the hardware has to provide as much computing power as possible and storage which does not limit the system that is planned in any way.

As the final step, the goal of the system is targeted, namely to be used by everyday mobile phone users. The software has to be designed to be user-friendly and, first and foremost, efficient. Even inexperienced users should be able to operate the system and use it after a short learning period. As it will act as an every day companion, it has to be as unobtrusive as possible and comply with the standard work flow of the smartphone. The software needs to be fast and intuitive to users in order not to be too disturbing even after long term use. Overall, it is crucial to work with user feedback to adjust the interface for pleasant everyday use.

A system like the one planned has to be extensible in order to push the limits even further. It has to be able to deal with external data sources like new sensors or other input so as to get input from additional information sources. This extensibility needs to be taken care of in the design process from the beginning on because it may be hard to implement this later. Bearing all this in mind, we will now start by exploring the market and selecting the best-fitting smartphone operating system.

4.2. Platform Selection

4.2.1. Introduction

A closer look at the smartphone market reveals that there is a variety of platforms available. Each platform has its own goals along with its pros and cons. This section gives a brief overview of smartphone operating systems available and highlights their specific features. The main attention is given to the usability for scientific research and development of applications in this field. As the smartphone market is growing and advancing rapidly, this can only be a snapshot of the current situation. The time of writing is late 2011, and the facts presented here will be outdated within a relatively short period of time.

4.2.2. Apple – iOS

Company	Apple
OS Family	Mac OS X/BSD/Unix-like
First Release	June 2007
Current Version	5.0.1
Programming Language	Objective-C
License	Proprietary EULA, some parts open source
Official Website	`http://apple.com/ios/`

The best-known smartphone worldwide is the iPhone running iOS by Apple. With their device, Apple revolutionized the mobile phone market and introduced several new terms like "apps" and the "App Store." These high-prized phones are status symbols for many people and are cult for a whole generation. The long-awaited launch of the first generation iPhone took place in June 2007. Followed by three generations, the latest addition to the mobile phone lineup was the iPhone 4s

Figure 4.1.: iOS logo by Apple

in 2011. Apple also is market leader in tablet computers with the iPad since its initial release in November 2010.

Apple is the manufacturer of these devices, and they are distributed either directly or by licensed partners such as brand stores and network carriers. Current mobile devices are the iPhone 4s, the iPad 2, and the iPod Touch 4G. The input method for all devices is a virtual keyboard. Key features are the well-designed housing and the excellent camera module for the iPhone. As a non-mobile device, there is a media center called Apple TV.

Programming for iOS is limited to computers running Mac OS X. The main software development environment is Xcode provided by Apple through the App Store. The iOS SDK contains everything for app development such as documentation and tools. The main programming language is Objective-C. This raises the entry bar for developers considerably. C/C++ is also supported but limited. In general, the developers are only allowed to use API calls documented by Apple. It is possible to root the device but then warranty is lost.

Figure 4.2.: iPod Touch 4G, iPad 2, iPhone 4s by Apple

The powerful API and the large developer community is a plus for iOS as a developing platform. For research use it is bound to the limits given by Apple with the API. This is a big constraint for most researchers. Additionally, the restriction to Mac OS X as the developing platform and the copy protection procedures do not give a lot of freedom when writing and testing apps. iOS is a great platform for commercial use but in general it is too restricted for the daily operations in scientific and research areas. The programming language does not allow rapid prototyping, which is crucial in order to test ideas quickly.

4.2.3. Google – Android

Company	Google, Open Handset Alliance
OS Family	Linux
First Release	September 2008
Current Version	4.0.2
Programming Language	Java, C/C++
License	Open Source
Official Website	`http://android.com`

Founded back in 2003, Android Inc. was bought by Google in 2005. The plan to create an open source mobile phone operating system was grounded with the founding of the *Open Handset Alliance*[1] in 2007. The first version of the Android software was 1.0 in 2008 accompanied by the first Android phone, the HTC Dream (also known as the G1). In the following three years Google published three major versions of their operating system, the current version is 4.0 (codename *Ice Cream Sandwich*). Android is the most used smartphone platform worldwide with over 500,000 activations every day in June 2011[2].

[1] `http://openhandsetalliance.com`
[2] `http://www.reuters.com/article/2011/06/28/us-google-android-idUSTRE75R31420110628`

As Android is open source and free software, any hardware manufacturer is allowed to use it in their hardware. This has led to a huge variety of available handsets. The main vendors are, alphabetically: HTC, LG, Motorola, Samsung, and Sony. There is every possible form factor available from large touchscreen devices to small phones with hardware keyboards. Android is known to run on cheap and slow hardware, too. This opened the market for phones available for the masses and low-budget devices. Google always offers a reference phone in cooperation with a selected manufacturer. The most recent addition is the Samsung Galaxy Nexus phone. The reference phone always has the latest software. Many manufacturers modify the operating system to fit their needs and add features and software of their own. Aside from smartphones, Android also runs on tablets. While Android 3 (codename *Honeycomb*) was designed for tablets only, Android 4 will reunite the operating system and will be running on phones as well.

Figure 4.3.: Android logo by Google

The main programming language for Android is Java. An independent virtual Java machine, called Dalvik-VM, processes the byte code of the running apps. The choice of this language offers the whole range of already existing Java libraries and programs. The API provided allows access to the built-in hardware, and nearly every existing Java code should be able to run on the phone with no or only slight modifications. For heavy calculation processes or games, developers can write parts or the complete application in C/C++. In general, there is no limit set by the API and the complete hardware is accessible. Normally, devices are locked and have to be rooted in order to gain full access to the complete file system. The rooting process is, as always, followed by the loss of warranty but some manufacturers have already begun to provide official unlock tools.

Many research projects relying on mobile devices use Android as the platform of choice. The openness and large community offers excellent support. Google itself promotes the platform for hacking and research applications and, therefore, offers API calls covering nearly the complete hardware. The development process is relatively fast which allows for software prototyping. The large catalog of already existing scientific libraries written in Java makes is easy to solve problems quickly and efficiently. Android provides a well-engineered rights management for applications which can be disabled on devices for development.

Figure 4.4.: Samsung Galaxy Nexus

Overall, Android is a good platform for research use. The programming language offers a variety of existing software and provides solutions for many problems. The huge hardware selection offers an application device for every need, and fast prototyping is possible as well as writing complex applications.

4.2.4. RIM – BlackBerry OS

Company	Research In Motion
OS Family	Proprietary
First Release	March 2002
Current Version	7.0
Programming Language	Java
License	Closed Source
Official Website	`http://us.blackberry.com/apps-software/`

Specialized in business clients, Research in Motion (RIM) offers BlackBerry smartphones with a highly secured environment. The phones are known for their ability to receive push notifications, for example emails and messages, but also contacts and more. The first BlackBerry phones were released in 2003. The latest generation is running BlackBerry OS 7. Aside from their mobile phones, there is a tablet OS called "BlackBerry Tablet OS" running the BlackBerry PlayBook only.

Figure 4.5.: BlackBerry OS logo

The latest hardware series are the Torch and the Bold devices. Although a key feature has always been the high quality hardware keyboard, the Torch series is the first offering a touchscreen only. The display is built-in in landscape mode and only the latest generation is equipped with a touchscreen. In general, the devices are high-priced and offer a good built quality.

Applications for BlackBerry OS are written in Java. Based on the Java Platform, Micro Edition (J2ME) BlackBerry provides an API which extends existing interfaces. This enables the developer to access the hardware on the devices. For advanced API access a certificate provided by BlackBerry is needed. This is used to prevent abuse in applications. The high security platform forbids any further access to the device than allowed by the given libraries. J2ME is limited and, in general, app development for this platform is relatively slow. This is one reason BlackBerry smartphones are not commonly used in research.

Figure 4.6.: BlackBerry 9900 Bold

The focus of BlackBerry is definitely business clients with high security needs. A strong security barrier comes along with a closed system and does not allow crossing any borders. This leads to a very specialized platform which does not fit the needs of scientific developing on a smartphone.

4.2.5. HP / Palm – webOS

Company	Hewlett Packard (prev. Palm)
OS Family	Linux
First Release	June 2009
Current Version	2.2.3 (phones) / 3.0.4 (tablets)
Programming Language	JavaScript / HTML5, C/C++
License	Palm EULA (proprietary), GPL
Official Website	`https://developer.palm.com`

HP webOS is the successor of the PDA operating system PalmOS. Founded in 1992, Palm was the market leader in personal information devices for many years. PalmOS was used in devices like the Palm Pilot, Palm V, and later in the Tungsten series. In 2009, Palm announced webOS as the new operating system for upcoming devices. The old operating system was outdated and not flexible enough to fit the requirements of future devices. The new webOS was said to revolutionize mobile experience and was developed from scratch. Palm was taken over by Hewlett Packard (HP) in 2010, the official name now is *HP webOS*. The development of hard- and software for webOS continued until late 2011.

Figure 4.7.: HP webOS logo

The only hardware vendor for webOS devices is the company Palm, later HP. The first device was the Palm Pre in 2009. Followed by several devices, there are generally two lines: One with high-end devices (Pre 1, Pre 2, and Pre 3) and a low budget line covered by the Palm Pixi and the HP Veer device. Every smartphone is equipped with a small hardware keyboard. In 2011, a tablet called the *HP TouchPad* was released.

The main programming language is JavaScript in combination with HTML5 and CSS3. This allows for the very fast programming of applications and a very high level of abstraction. For direct access to devices like GPS, which is not available in standard HTML specifications, a system-bus API is available. For individual needs these services can be written in node.js[3]. For games and computationally intensive tasks, C/C++ can be used. SDL is available for graphics and openAL for sound . In general, there is a full Linux environment with standard libraries under the hood. There is no need to root devices. A developer mode can be turned on with full access to the underlying Linux system. A homebrew community[4] provides system modifications and patches.

Figure 4.8.: HP Veer, HP Pre3, HP TouchPad

On the one hand, it is possible to build prototypes very quickly due to the level of high abstraction. Creating applications in JavaScript and HTML is very comfortable but also

[3] `http://nodejs.org`
[4] `http://webos-internals.org`

limited. On the other hand, complex computations and direct access to devices like media and camera hardware are not available. Even when using C/C++, there is not a full API for the entire built-in hardware. This limitation can be avoided by using unsupported Linux APIs. The platform is great for creating fast user-interface testing studies, yet for media and sensor intensive applications this is not ideal. A plus is the standard Linux, which has been used as basis for the operating system. This makes it easy to port existing projects to the device. In some special cases it might be the right platform, but overall it is too limited in its hardware access, especially if someone does not like to implement it all from scratch.

As a relatively new platform webOS had the chance of becoming a competitor to the existing platforms. The marketing strategy and nebulous decisions of Hewlett Packard led to a loss of customers and prevented a further growing of the distribution. But in some cases it is the ideal research platform to create prototypes and test ideas without spending hours of coding. The abstract art of programming allows the developers to implement and test ideas rapidly.

4.2.6. Microsoft – Windows Phone 7

Company	Microsoft Corporation
OS Family	Windows CE
First Release	November 2010
Current Version	7.10 ("Mango")
Programming Language	Silverlight, C#
License	Proprietary (Microsoft EULA)
Official Website	`http://create.msdn.com`

The latest addition to modern smartphone operating systems is brought on by Microsoft. The successor of Windows Mobile is called Windows Phone 7 and was built completely from scratch. With this new platform, Microsoft aims at the consumer market and offers a design which is silhouetted against the competitors. The Metro GUI is based on so-called "Live-Tiles" which are squares or rectangles dynamically displaying information to the user. The whole system aims at being user-oriented and converging important information on the main screen.

Figure 4.9.: WP7 logo

Windows Phone 7 runs on devices created by selected manufacturers. It has to be licensed and at the moment there are ASUS, Dell, HTC, LG, and Samsung devices available on the market. The latest cooperation is with Nokia promoting the Lumia 800 phone. Microsoft forces relatively tough minimum system requirements on the licensees. This includes a WVGA resolution screen, a fast CPU and GPU, and several specified sensors. This results in a very homogeneous lineup of devices.

Figure 4.10.: Nokia Lumia 800

Microsoft provides the Windows Phone 7 Software Development Kit without any fees. It requires Microsoft

Windows and integrates into Microsoft Visual Studio. Applications use a specified version of Silverlight or XNA where the main programming language is C#. Developer phones have to be unlocked before third party applications can be installed. There is a large set of tools for designing and creating mobile applications for Windows Phone 7 available by Microsoft. This platform is a candidate for rapidly creating prototypes for experiments. As the operating system is novel to the market, the available API has not yet been fully completed. Some functions that allow access to available hardware are missing.

The Windows Phone 7 platform has the potential of becoming a good alternative to Android in research areas. The largest issue is the commitment to Microsoft Windows as the developing operating system as well as the fact that it is a very closed system. There are many Microsoft programmers available worldwide which help the platform grow very quickly. The re-skill process is very short, this has led to a quickly growing number of applications available for Windows Phone 7. Generally speaking, time will have to show how it develops further, but there is a positive outlook.

4.2.7. Other Mobile Platforms

Aside from the main players, there are several other smartphone platforms which run on a minority of devices. There is Meego[5] by Nokia, which offers a full open source stack. It is designed to run on smartphones and netbooks. Inconsistencies within the management team have led to the official discontinuation of the software. A project called Tizen is aimed at the continuation the development. The latest phone using Meego is the Nokia N9, which is nearly identical to the Nokia Lumia 800 using Windows Phone 7.

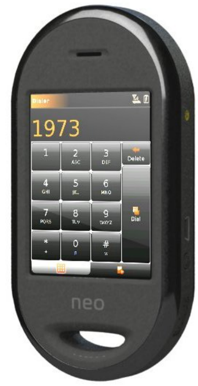


Figure 4.11.: FreeRunner open source mobile phone

There is a small group of people who follow the full open source idea with an open source operating system but also an open source handset. This project is called Openmoko[6], which had its first release in larger quantity with the Openmoko FreeRunner in 2008. There are several distributions running on the device but none of them is stable enough for everyday use. This is the only existing project with open source hardware.

The diversity of smartphone platforms has brought several cross platform frameworks to life. A well-known ambassador is PhoneGap[7]. PhoneGap supports seven platforms including all the major ones presented in this chapter. Applications are written in HTML5 and JavaScript. The framework encapsulates the native API and provides standardized access. The major drawback of PhoneGap is that it is relatively slow and does not cover the full range of available API calls of a specific platform. Another problem is the look and feel of applications. As these frameworks do not use the native UI elements, the developer has to provide a look-a-like design for each platform and supply a completely different design. In general, PhoneGap works well for applications that are not too complex.

[5] http://meego.com
[6] http://openmoko.org
[7] http://phonegap.com

4.2.8. Conclusion

The main requirement for a mobile operating system used in this project is that it must be adaptable for research use. The software has to be a prototype showing the functionality of the proposed system. Therefore, it must be able to fulfill all the tasks plus, it should not take a lot of time to develop applications for it. The desktop environment used is based on Linux, which enforces a platform that supports Linux as development operating system. In order to get a maximum of support for upcoming problems, one of the main platforms has to be chosen. This will likely guarantee that the project does not run into a dead end.

Windows Phone 7 and iOS are bound to their desktop operating systems Windows and MacOS. BlackBerry OS is too limited in its hardware access and does not offer the freedom needed in terms of functionality. Taking this into account, webOS and Android are the only two systems left to consider. Any kind of developing for webOS is very high-level and abstract. For researching user interfaces or user behavior on smartphones it may be a perfect match, but it limits the developer in direct access to hardware and sensors.

The best tradeoff in terms of prototyping abilities, hardware access, community size, and end user distribution is Android by Google. There are many different devices available on the market fitting every need, and Google actively updates as well as continues the development of the operating system. Java as programming language offers a huge community, relatively fast prototyping abilities and a library for any existing problem available. The Android API offers high-level access to the smartphone hardware but also does not lack direct access to information and sensors. The current Android version for phones is 2.2.X (codename *Froyo*).

4.3. Hardware Requirements and Selection of a Suitable Smartphone

4.3.1. Overview

The Android operating system is built completely independently from hardware manufacturers. It is designed to run on a large variety of form factors and hardware platforms. In order to show the latest additions and features to the system, Google always cooperates with a manufacturer when developing the next version of Android. Together, a reference phone is built for the release version. When the version is officially made public, the source code is available to everyone under an open source license.

From low budget to high-end phone and from brick to slider designs with or without physical keyboard, every conceivable configuration of a smartphone is available. At the moment there are several hundreds of handsets on the market. The main differences aside from the case design are processing power and display resolution. Google requires a minimum configuration to allow handsets to access the official Android Market. This includes a GSM module and a camera.

4.3.2. Requirements

The two main requirements for the desired handset are computing power and built-in available sensors. The phone should be able to handle the given tasks easily allowing for room for later additions and extensions or parallel running processes. A fast dual-core processor and a large working memory (RAM) are ideal. Persistent storage media must be exchangeable and extendable in order to fit upcoming needs; further, a minimum of 2GB is required to store the accumulated data in later experiments.

On the sensor side, it is important to be able to access the user's current location. This implies a GPS sensor which is standard for all modern smartphones. A frequent query for the current location will take place in regular intervals, and for this reason the sensor should be able to provide accurate location data very fast. The latest GPS receivers offer a feature called Assisted GPS. A-GPS uses location information from other sensors like the cell tower location in order to support a very quick first GPS location fix. A 3-axis accelerometer is standard in getting the current orientation of the device. A gyroscope is nice to have for the retrieval of even more precise and complete information in 3D space.

The phone used for this study will be given out to subjects for testing, which also means that it should be very robust. The typical brick design with a well-built housing in this case is ideal.

4.3.3. Selection

In January 2010, Google released the latest reference phone for Android version 2.1, the Google Nexus One. The hardware partner for this phone was HTC. As the reference phones always implement the latest additions and features of the Android platform, it is the perfect choice for the study. Additionally, the past has shown that Google provides the upcoming operating system updates.

The Nexus One was not available for sale in Germany at the time of this study, but HTC offered a nearly identical model called HTC Desire. The HTC Desire has a slightly different design: The trackball was replaced by an optical counterpart, but overall the specifications are the same. With the HTC Desire there is a high-end latest generation Android smartphone available for this study. It was chosen as the testing platform for upcoming studies.

Mobile dual-core processors are not available for smartphones at this time, but the HTC Desire offers a high-clocked 1GHz processor with 576MB RAM. The storage is expandable by Micro-SD-cards, it ships with a 4GB card. A-GPS is available but no gyroscope. There is no Android phone with a gyroscope available at the time of writing, which means that is no alternative at this time. The housing is very solid consisting of a *Gorilla Glass* front with metal framing. The battery is covered by a plastic back with a rubber surface.

4.3.4. Model Specifications and Sensors

The HTC Desire (see figure 4.12) was released on February 16, 2010. The first release has an AMOLED display built-in. Later in 2010 it was exchanged for a Super-LCD panel. The rest of the hardware remained the same. The general specifications are:

Figure 4.12.: HTC Desire with HTC Sense User Interface

Manufacturer	HTC Corporation
Type and Form Factor	Smartphone / Slate
First Release	16 February 2010
Operating System	Android 2.1 (Eclair), later 2.2 (Froyo)
Dimensions	119mm x 60mm x 11.9mm
CPU	1 GHz Qualcomm QSD8250 Snapdragon
GPU	Adreno 200 (AMD Z430)
Memory	576 MB RAM
Storage	512 MB flash
Removable Storage	microSDHC up to 32 GB
Battery	1400 mAh Li-ion
Display	3.7-inch 480x800 AMOLED / Super LCD

The following sensors are built-in and accessible by the Android SDK:

- **Microphone:** As a standard component for a mobile phone, the microphone can be accessed directly. This enables the recording of surrounding sounds.
- **Accelerometer:** The HTC Desire includes an embedded three-axis accelerometer. The main usage is to rotate the screen automatically. In some games the orientation in reference to the ground is used as input, e.g. for steering a vehicle. The raw values in multiples of g-force are available.
- **Magnetometer:** Normally used as a compass, the magnetometer supports the accelerometer in defining the current orientation of the device.
- **Light sensor:** It is placed at the top of the display and in normal operation it adjusts the screen brightness. The raw values are available.

- **Proximity sensor:** Aside from the light sensor, there is a short-range proximity sensor. The typical use is to disable the screen during calls when the phone is near the ear. This prevents unwanted touch events on the screen.
- **Bluetooth:** Bluetooth enables the device to discover and connect with other Bluetooth enabled devices in its close proximity. Several Bluetooth prototools, like for file-sharing or serial connections are pre-installed.
- **Wi-Fi:** The main task for the Wi-Fi module is to connect to wireless networks. Aside from this, it can be used for locating the device. All surrounding SSIDs and MAC-addresses are sent to a service provided by Google, which responds with an estimated location. The accuracy heavily depends on the area and is not available everywhere. Every user of this method submits the correct position automatically to the server when available via GPS. This improves the location database over time.
- **GPS:** Worldwide localization, height, and speed are provided by the Global Positioning System. Using satellites, the position of the receiver can be calculated with a deviation of about 10 meters. The HTC Desire features A-GPS for faster localization.

The phone is equipped with two speakers, one at the front typically for hearing someone when calling and a louder one at the back for hands-free talking and loud acoustic signals. A vibration motor and a colored LED can be used for silent attention.

4.4. Considerations for Software / User Interfaces

4.4.1. User Interface

The goal is to create software which can be used as a constant companion for the user. The user will interact several times a day with the device. For this reason, the interface has to be very goal-oriented and clean. When recording daily activities, the typical interaction time will be very short, for example only a few seconds. This calls for very little to no loading or waiting time.

The software has to work reliably and should never corrupt existing data. This in turn leads to some slower than possible processes but, on the other hand, ensures that after data corruption at least remaining parts are available for analysis. The application should be able to detect problems and resolve them automatically. Only in drastic situations should it notify the user to take manual action. One possible solution for this problem is to separate the application in small logical parts which operate individually.

A clean user interface makes use of known paradigms like the magical number seven for interface options. Miller states that a human can handle a maximum of 7 ± 2 chunks of information in their working memory [63]. Here, this applies to information as well as buttons on the screen. If there is a need for a larger list of options, they have to be grouped. Icons guide the user in finding the corresponding action faster.

Before the first user studies can take place, the user interface and application flow have to be tested with several end-users. It is important to select testers who are not technophile only because in the long run the application is targeted at a broad audience.

4.4.2. Implementation

When developing a mobile application, there always is the question as to whether it should be implemented with the native development kit or whether it is possible to use a cross platform framework in order to address a larger user-base at a later stage. PhoneGap was introduced in the last chapter as one framework for multi-system development. PhoneGap is currently available for the following platforms and supports the listed API calls on in the left column:

	iOS	Android	BlackBerry	webOS	WP7
Accelerometer	X	X	X	X	X
Camera	X	X	X	X	X
Compass	X	X			X
Contacts	X	X	X		X
File	X	X	X		X
Geolocation	X	X	X	X	X
Media	X	X			X
Network	X	X	X	X	X
Notifications	X	X	X	X	X
Storage	X	X	X	X	X

At first glance this looks very promising. Android, as the targeted main system, and all other major platforms seem to be fully or mostly supported. The main problem is that PhoneGap only provides API access to the most common interfaces. Additionally, the access is very high-level and abstracts the original interface to fit all platforms. The general rule is that the PhoneGap interface provides the common ground for all supported operating systems. This restricts the access and does not allow direct manipulation and information retrieval of specific values. For example, PhoneGap does not allow access to detailed information about the available Wi-Fi networks and completely lacks access to the Bluetooth adapter.

For a research project like the one planned, this is a deal breaker. The system will rely on sensor values and it is not reasonable to miss information because of the restricted and abstracted API. PhoneGap is extendable with modules of its own which can be adapted to the individual platform and which could provide the missing information. But this is very labor-intensive for each targeted platform.

Also, PhoneGap, as well as most cross platform frameworks, uses HTML5 in combination with JavaScript as its programming language. The resulting web application is embedded in a *webview* container. JavaScript performance has improved very much in the last years but is still not at the level of compiled or byte code using languages. Especially when computing power is needed, JavaScript most certainly is the bottleneck of the project.

For a research project which is supposed to produce a prototype not targeting as many platforms as possible, a native application is the best choice. This can be achieved by choosing the best fitting platform before the complete range of available interfaces can be accessed. Android offers a low-level API for direct access to sensors and built-in hardware.

The main programming language for the Android SDK is Java. Java enables the use of a variety of existing libraries and runs efficiently with the Android Dalvik virtual ma-

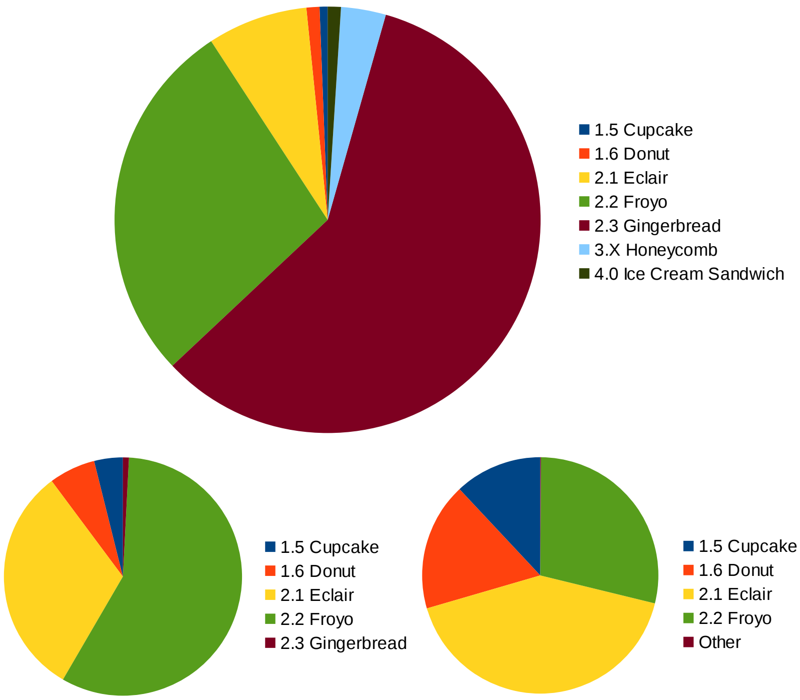


Figure 4.13.: Android system version fragmentation in 02/2012 (top), 02/2011 (bottom left), and 09/2010 (bottom right). Source: Android Developers [24]

chine. For very performance-critical portions of the application, Android offers a so-called native development kit (NDK). The NDK extends the normal SDK by adding C/C++ as programming language to the application. The code can be integrated into the application and accelerate complicated computations. Normally, the NDK is used to build graphic-intensive applications like games offering access to 3D openGL function calls.

For a long time Android has been suffering from high fragmentation regarding the version. In the last years the majority of phones upgraded to Android 2.X. In 2012 more than 85% were running Android 2.2 or higher. The new versions bring along new APIs and many other enhancements. Taking these figures into consideration, a phone application should be based on Android 2.2 in order to support the majority of phones in the wild. The selected HTC Desire has officially been updated to Android 2.2, which means that this will be the code base for the project. This offers a large user-base for later public testing but also does not restrict the development too much.

The official Android SDK offers the best support for the hardware. For the ongoing project there should not be any limitations by the API. For even more direct access to the hardware and software components, it is even possible to access the underlying Linux system. There is unofficial rooting software available allowing root access. With a rooted phone, every part of the system can be modified and customized. It is possible to extend the core system and use self-compiled kernel modules.

For the HTC Desire there is a fan based Android distribution available called CyanogenMod[8]. The modification features the latest Android version even if the phone is not offi-

[8] http://cyanogenmod.com

cially supported. Also, back-ports of features and self-contributed additions are available. In general, rooting the phone voids the warranty but in the meantime several hardware manufacturers have been starting to official rooting software.

Overall, the presented methods should not imply any show stopper. It is just a question with how many phones the application will be compatible. For a start, the official Android SDK with its already extensive functions should be sufficient. The goal is to build software that fulfills the researching aspect of this project, but when possible it should also be a goal to reach a broad base of Android users.

4.5. Thoughts on System Limitations & Resource Efficiency

The decision to use a common smartphone as the hardware basis, forces the project to deal with the platform provided including all its drawbacks. It is not a platform specifically designed for this project, so it does not fit perfectly. The built-in sensors were not constructed for scientific use and, therefore, are not highly precise instruments. The price is always a very important factor when creating a commercial product because the final product has to be as inexpensive as possible. Apart from these limitations, it is a huge advantage to use a well-designed platform for this project that is actually available on the market. The whole designing and testing work of such a device would exceed the time frame of this work. In the end it is easier to accept and deal with the limitations that come with a commercial phone.

4.5.1. Accuracy of Phone Sensors

In order to determine the current position of the phone via GPS, the sensor receives signals from satellites. Roughly speaking, each GPS satellite sends its current position and a time stamp in a 1500 bit long package. The data rate is 50bit/s and the receiver needs the full packages of a minimum of three satellites for the calculation of the current position by triangulation. In the worst case, the receiver has to wait up to 30 seconds until the next message block begins.

The accuracy of the resulting position depends on the number of accessible satellites. The surrounding area is also important because the GPS is very inaccurate for positions where, for example, large buildings reflect and that way adulterate the signal. Bad weather and clouds can also lead to no or a very weak signal. A fact that affects especially mobile phones is that the device is very small and the users could possibly cover the GPS antenna with their hands. In principle, the best position estimation is available with clear skies and nothing covering the phone. Under these conditions a precision of under 10 meters is possible.

In a typical scenario it takes 15-30 seconds for the first position fix which has an accuracy of about 50 meters. The GPS module uses additional information from the last signals received in addition to a web service for faster locating. Over time more satellite signals are available and the position becomes more accurate down to 10-20 meters. The current accuracy can be calculated by the GPS module, and the value can be accessed by the software.

In order to avoid the relatively long time up to the first position fix, the Wi-Fi module is used for a first estimate. As already said, here the accuracy depends heavily on the data available. In contrast to a GPS position, here the accuracy is not known. The software can estimate it but this data is not reliable. There are several providers for Wi-Fi positioning available on the market, such as Skyhook[9]. Android uses Google's own service. By accessing the service, the users agree to submit their own positioning information in order to update and complete the database. This process is active even when the user does not require the current location and it runs in background. The latest development by Google's service is an opt-out function for access point owners not wanting their devices in the database. By simply adding the suffix "_nomap" to the SSID the access point will not be added to the database. This is also convenient for non-stationary mobile hotspots like smartphones used for internet sharing.

It is crucial to know the source of positioning information in order to deal with it correctly. Also getting a reliable position takes some time, which should be kept in mind when defining the time frame for reading the sensors. The amount of time and the overall accuracy depend on the mobile phone used.

Many Android phones, including the HTC Desire, have a magnetic compass. The hardware providing the data is a triple axis magnetometer. The precision varies from phone to phone and does not always work reliably. Also, after some time of use the module has to be calibrated depending on the current location. This leads only to rough information on the orientation.

In summary it can be said that smartphone sensors are not built to deliver highly precise data. They were built with the intention of satisfying the default operations like rotating the screen or rotating the map of an application. For a research project it is important to keep this in mind and facilitate the handling of measurement errors correctly.

4.5.2. Limitations of the Phone

The battery of the HTC Desire is rated with 1400 mAh at 3.7V. The manufacturer information specifies 340 hours of standby and a maximum of 6.5 hours talk time. The typical operation in this project will be very power intensive and the phone should at least provide power for one full day. The manufacturer specifications should allow this without problems but when using all sensors, especially the GPS module, the battery will drain much faster. This in mind, the application has to handle the amount and length of sensor readings carefully to still fulfill the requirement for at last one full day.

Figure 4.14.: Extended Battery Cover by Mugen Batteries, photo by tracyandmatt[10]

Another approach would be to extend run time by using a battery with a higher capacity. In general there are two types of models available. The first one is a simple replacement of the battery, which sometimes offers a slightly higher capacity. For the HTC Desire there is a 1600 mAh version[11]. If there is a need for even more capacity, there are batteries

[9] http://skyhookwireless.com
[10] http://www.tracyandmatt.co.uk
[11] http://www.mugen-power-batteries.com/mugen-power-extended-battery-for-for-htc-desire-

available offering up to 3200 mAh[12]. These have the drawback of not fitting into the standard housing of the phone. It comes with an alternative back-cover which holds the larger battery as seen in figure 4.14.

Android has been designed to run on smartphones that are able to connect to the internet regardless of whether there is Wi-Fi available or not. Many services like Wi-Fi positioning require a connection to an online server. In the study the users will not configure the phone as their primary one so there is no connection when not logged onto their Wi-Fi. This could influence the accuracy and availability of at least the localization.

Another drawback is that the software design of Android only fits the typical use-cases. For example, when the Wi-Fi API is used to scan available access points regularly, it sometimes happens that the phone loses its connection to the current network and does not reconnect. The same applies for some other APIs like Bluetooth. The application has to work around these problems to ensure a normal operation of the phone. This is important in case the users should use it as their primary phone. The application should run silently in the background.

Very often the memory cards provided are very cheap and usually not high quality products. They are probably break when used heavily like in the case of many reading and writing operations. This should be tested before the field test and perhaps the cards should be replaced by high-end products. This could circumvent possible frustrations because of data loss or corruption during a user study.

4.5.3. Resource Efficiency

During the user study, the phone should last at least one complete day to enable disruption-free user operation. The phone can be charged over night, which is comfortable, as it is supposedly not used then. An analysis of energy consumption of the smartphone's internal hardware is necessary to ensure the targeted run time. With this knowledge, the application can be designed to be as energy efficient as possible.

Carroll and Heiser analyzed the power consumption of a smartphone in depth [18]. They took an Openmoko FreeRunner phone for detailed examination and compared it to the HTC Dream and the Google Nexus One phones. The open hardware design of the FreeRunner made a very precise measurement of the power consumption possible by placing measuring points at the circuit level. A small drawback here was the outdated hardware which had been used, but nonetheless it should be transferable to current smartphones.

The typical power consumption of a phone is shown in figure 4.15 on the left split up into the internal components. When the CPU and RAM are busy, they will leave the power-saving mode and increase the clock frequency. This leads to a combined maximum power consumption of up to 200 mW/h. The largest battery drainer is the display with its backlight. When the backlight is switched on, it consumes up to 400 mW/h of energy. The interesting part here is that the intensity of the backlight seems not to be the linear equivalent of the current drain as seen in figure 4.15 on the right.

When using additional hardware, more power will be used. While the GPS consumes

htc-bravo-htc-a8181-htc-epic-t-mobile-htc-desire-vodafone-htc-desire-1600mah.html

[12] http://www.mugen-power-batteries.com/mugen-power-extended-battery-for-htc-desire-a8181-with-battery-cover-in-brown-3200mah.html

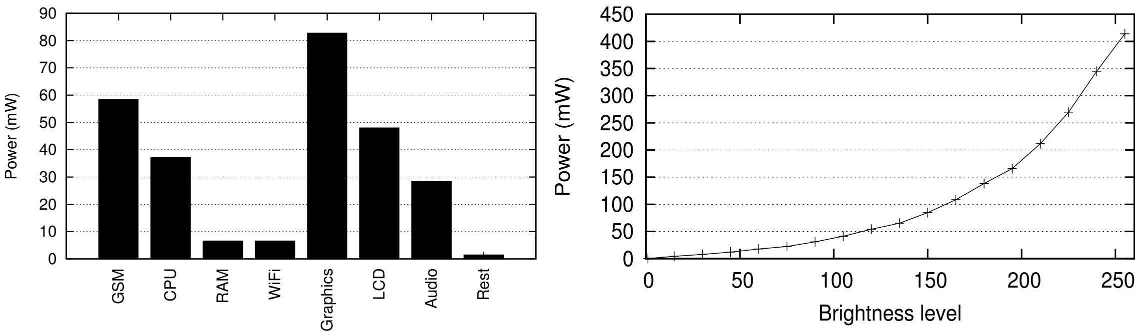


Figure 4.15.: Left: Average power consumption of a smartphone while in idle state with backlight off. Aggregate power is 268.8 mW/h. Right: Display backlight power for varying brightness levels. Source: Carrol [18]

about 150 mW/h, the wireless communication modules request a huge amount of energy when active. Both GPRS and Wi-Fi each need over 600 mW/h of power while transferring data. This is as much as the complete system needs for running. Possibly, the usage time of these modules should be kept at a minimum. Bluetooth is a relatively low-energy device consuming about 35 mW/h.

Because the individual hardware modules consume a significant portion of the overall power, several pieces of work exist that reduce this by intelligently activating and deactivating single parts. The main idea is to use low-energy sensors like the accelerometer to detect the need of additional sensors. The system has to decide if the current situation can be detected properly if there is a need for more information. Wang et al. show in their study that in their scenario an increase of battery life of more than 75% is possible [91]. A similar approach has been presented by Abdesslem et al. [10].

A lot of energy saving work is already implemented in the Android subsystem as unneeded modules are deactivated automatically when not in use, and also the kernel decreases the CPU's frequency when the device is idle. For the ongoing project the goal for the battery is to last at least one full day and, in order to achieve this, the energy saving tips can be used. It remains to be seen if an intelligent system for managing the sensors is necessary, but it is a good start to reduce interaction time with the user to a minimum as the display is one of the main power consumers. In summary, it is good to know where energy can be saved to create an energy-efficient and long lasting system.

4.6. Extending the System

A system only using the originally provided smartphone hardware will in some cases reach its limitations. For this is reason it is important to provide the possibility of extending it by advanced sensors and other input. This will allow the system to detect more complex situations and activities. Here are some examples for suitable extensions.

4.6.1. Smart Environments

The environment we live in is turning more and more into a technologically enhanced surrounding. Houses are equipped with sensors for heating or power control and the interior is becoming smart by built-in sensors and displays. In the future we will live in so-called smart environments. According to Mark Weiser, a smart environment is “a physical world that is richly and invisibly interwoven with sensors, actuators, displays, and computational elements, embedded seamlessly in the everyday objects of our lives, and connected through a continuous network” [22].

A smartphone could be connected to this network and get access to all the aggregated information. A smart home will provide data about the room usage and the presence of people. It knows when someone enters a room and where people are inside. The refrigerator and the kitchen are ”aware” of the meals of the members of the household, and the bed provides information about sleeping patterns. Also the quality of sleep can be measured and fed into the network.

All this information is very useful for an activity recognition system, and in future the smartphone could get access to these sensors and stored information of a smart environment. Some projects like the SmartHome Paderborn[13] are a first start to bring this technology to the normal users, but at the time being there is no commercial comprehensive system available. Specialized products for individual needs like the body scale by Withings[14] are a step in this direction. Withings offers a Wi-Fi connected scale which uploads the weight to a web service which aggregates and provides the preprocessed data for third party services.

4.6.2. Computer / TV Usage

In Germany, television audiences are observed in their TV watching behavior, i.e. who and how many people are watching what exactly on their TV. This is done by a special device given to a selected group of families whose members are supposed to reflect the typical television users. This box is owned by a company which provides the resulting data. The device is able to collect data about the current broadcasting station and monitors the time spent watching each channel. Each family member has to log in before starting to watch TV allowing the usage to be assigned to the individual person or family member. The collected data are sent to a central server every morning.

While this is very worthwhile data for the broadcasting stations, it is really interesting for activity tracking and evaluation, too. The data could be used to determine the mood of the user by interpreting the shows watched. Also the general daily usage could be a good indicator for activities. In the future, internet enabled TV could replace the additional box and could enter the collected data into the smart environment network. This way it will be available to the smartphone just as an additional sensor. As television is an important part of many peoples’ lives, it is crucial to know details about its consumption. It will give a deeper insight into the daily routines of the user.

For personal computers there are several software tools available which monitor the usage

[13] `http://smarthomepaderborn.de`
[14] `http://withings.com`

of the user. With the already existing user management, the activities can be assigned and the software only has to track which programs are used. For some programs, like the browser, it makes sense to save additional usage details like the current website. One example is the software RescueTime[15]: It claims to monitor productivity while at the computer and gives detailed statistics when using which software. Additionally, it can be configured to interfere the current actions of the user by blocking websites manually defined or automatically determined by the program.

Again, the main use might not be integrated into a smart environment but the software shows that there is the possibility of getting detailed usage data for a specialized device. Like television it gives an insight into the use of a computer. While the television is normally used in spare-time only, the computer is also a tool many people interact with at work. A detailed view of the activities while in front of the monitor is available and will enrich the daily activity data.

Both, television and the computer are good examples for an activity where a smartphone itself cannot go into detail. The detailed information is just not accessible by the device and has to be provided by an external source. For electronic devices it is easier to collect these data and submit them to the smart environment system where they can be used as a sensor for the system and help to detect the current situation more precisely.

4.7. Summary

The chapter started with a detailed analysis of the requirements of the planned system. This was the groundwork for the ongoing decisions in terms of platform, hardware, and software. A flexible, prototyping-enabled and robust basis is needed and this was found in Google's Android operating system. All the other platforms have shown drawbacks in the development process, the access to the sensor data, or hardware available. For Android there is a huge selection of phones available but in the end it was clear that a high-end model is required to avoid possible bottlenecks posed by the hardware later on. The "Desire" phone by HTC was selected for this study as it offers high computing power, a nearly complete set of sensors, and a robust case design. It is also identical in hardware to the Google reference phone "Nexus One," which is known to be supported very well in upcoming software updates.

Some notes about the user interface and its requirements were discussed and also further implementation details. A platform independent framework was considered but cannot be used as its strong abstraction layer does not allow direct access to sensor information, which is definitely needed for this project. This led to the decision to use the Android Java SDK, which offers a huge collection of ready-to-use available libraries but also enables relatively fast and efficient prototyping for the development process. To cover the majority of available Android devices, the minimum platform version was set to version 2.2 or higher which run on over 85% of all Android phones.

When developing a mobile system running with battery power, a look at the limitations and resource efficiency was inevitable. Notable facts on sensor accuracy were discussed as well as the main limitation of each mobile device: the battery life. The last part was about

[15] `http://rescuetime.com`

how the system can be extended by additional sensors which may be used to complete information which cannot be provided by the phone itself. This could become necessary, and it is crucial to plan this beforehand.

The next step is to get hands-on experience with devices and develop an Android application which collects first data. These data then will be the groundwork for further investigation and will give an insight into what can be expected as output from the system. Subsequently, the next chapter is about the first user study with the main goal to obtain data to work with.

5. Development of a System to Collect Daily Activity Data

This chapter is about the first of two user studies. It includes the preparation, procedure, and partial evaluation of the collected data. This user study is the first step towards collecting sensor and label data to work on in order to get a general overview over daily activity data and hints for later classification approaches. It concludes with the evaluation of questionnaires handed out to the participants. The resulting data itself will be inspected separately in the next chapter.

5.1. User Study Motivation and Goals

A suitable set of representative example data has to be evaluated first in order to fulfill the goal of this project, namely being able to conclude daily activities (d.a.) from sensor data. There are already some projects which offer publicly available data sets like the one by Microsoft Research[1] or the one by Zheng et al. [99], which contain GPS location data only. There are even sets with multiple sensor data like the "Lausanne data collection campaign" project [48], which includes accelerometer and phone usage data. The main problem of all these data sets is that the primary information is missing: the labels for the corresponding sensor data that include daily activity information. All projects aim at a general analysis of the collected data and for this reason do not fit the problem of this project. In conclusion, newly labeled data have to be recorded so that both, sensor data and corresponding activity labels are available.

The best results for later processing can be achieved by working on realistic, real-life data. Further, it is important that the sensor information is collected manually by people actually carrying the targeted phone with them. A study has to be planned in such a way that it provides adequate and labeled sensor data that can then be worked with. The goal of this study is to acquire data sets of several individuals who fit the needs of this project as much as possible. This study is the first of two studies in an iterative process to help create the planned system, which will then finally be able to support users in activity tracking. The recorded and labeled data will give a first insight into what sensor data look like and the correlation between these data and activity labels. Also, a first analysis and evaluation can be done using the collected information.

[1] `http://research.microsoft.com/en-us/downloads/b16d359d-d164-469e-9fd4-daa38f2b2e13/default.aspx`

5.1.1. Upcoming Challenges

First of all, it has to be determined how people define daily activities and what activities are important to them. The definition of what an activity is, is highly individual and depends not only but also on how people structure their day. Also, a time frame for the study has to be set. The idea is to use these data sets for a generalization, but this means that there has to be a sufficient amount of data for each activity label per individual. On the other hand, the requirement to get real-life data from people restricts the time for the study, because participants have to be paid and really use the data collection system. A good trade-off has to be found here. During the study another time-related variable is the density of data recordings. The battery life of the smartphone limits the recording time, which in turn means that the interval and length for sensor data recording have to be defined. These settings have to provide valuable sensor data reflecting the current activity but also have to respect the power consumption of the phone. Along with the sensor data, activity labels have to be recorded. Here it is important to find an acceptable time interval for the users to integrate into their current activities.

The whole process of data recording and activity label entering has to be implemented in an application which runs on a smartphone. Aside from the technical challenges, the application has to be easily useable by the participants. There is need for a simple, efficient but complete user interface. On the back end, the application has to record and store data reliably within the given configuration. Another aspect is to find out how the participants work with the system itself. During the study, the participants' interaction cannot be observed properly and a follow-up questionnaire is a good way of evaluating the system. These questionnaires have to be designed carefully in order to provide data about the system itself as well as about the users and their perception of the system. Additionally, questionnaires can provide possible enhancements suggested by the participants for further development of the next run-through of the system. Overall, a well-designed and fail-save application has to be built for the on-going study.

5.2. Experiment Design

A complete user-customizable list of activities is very hard to evaluate due to a different labeling of the same activities by different users. Therefore, a list of proposed activities should be available to the users. This list should cover as many activities as possible and only exceptional items should have to be entered manually by the users. A preliminary short questionnaire will be conducted in order to build a common list of activities and corresponding categories. The results will give a rough idea which activity labels and categories have to be pre-set. In this questionnaire daily activities have to be named and grouped into categories. The categories will help to organize all activities and provide another level of abstraction. The results of this preliminary study will be the basis for the first data-collecting study.

Some preliminary testing has been done to determine the best configuration of the timing of actions for the application. First, the length of each sensor recording was defined. The minimum time is given by a technical constraint: The GPS sensor needs a minimum amount of time to return reliable and precise position information. This time is about 15 seconds. For more than one location fix the time for sensor data recording is set at

20 seconds. All other sensors are able to collect sufficient and meaningful data in this time frame. The interval of recordings is mainly limited by the battery life of the device. The smartphone should be able to run at least one full day without having to be charged. Previous tests have shown that is not possible to record data for 20 seconds any more often than every 3 minutes. A good time interval for interrupting and asking the users for their activities has been found by pre-tests to be 15 minutes. During normal operation, the application will record data continuously. As there might be a need for privacy time during the study, the application will have a mode in which no data is recorded. This also applies at night when the participants are asleep because the phone will only provide useable data when carried on-body. Limited by the money available for the participants, the study will last 14 days with 16 people. This covers two complete weeks and should provide enough generalizable data.

The goal of the experiment is to collect labeled sensor data for a longer period of time. The users have to carry around the device all day long and have to enter their current activities within a fixed time interval. This provides labels for the sensor data that are recorded automatically and continuously in the background. It is important to keep in mind that because the study will last several days, the experiment and the application have to be designed in such a way that is pleasant to participate and, in turn, ensure that the subjects will not stop using the device correctly or only in random intervals as a result of an either bad or complex interface design. Furthermore, it is important to get complete daily activity data for later processing. There is a variety of different activities which have to be organized in such a way that the entry process is quick and supports the user in finding the right activity. This is achieved by grouping the activities into categories. Each activity belongs to a specific category which also provides hierarchical information. The time for the users to pick a specific activity can be decreased dramatically by this categorization. A fixed grid view of categories and activities additionally increases the efficiency as the user can rely on the position of already known activities. This implementation fastens the process of selecting a specific item. As a side task the users will be asked to enter their current mood on a scale from 0 to 100. This is an interesting additional value to compare with the daily activities and sensor data.

A questionnaire before and after the main study will provide more information about the subjects. Aside from general information like age and sex, it is interesting to know more about the participants' previous knowledge of using smartphones, and what they expect from such an activity diary system. They also can guess which activities they will be doing during the study. These results could be compared to the entered activities later on.

Finally, these concepts have to be implemented into an application for the Android smartphone. The application will have to collect sensor data in the background and has to record the users' input. It will be the only study companion and will hold all the collected data. In order to provide stability and reliability, all components of the application are possible, both separately as well as combined. Crashes of the application must not lead to data loss or stop the recording of sensor data. As mentioned before, a preliminary questionnaire will provide the last piece of missing information: a common collection of activity labels and categories.

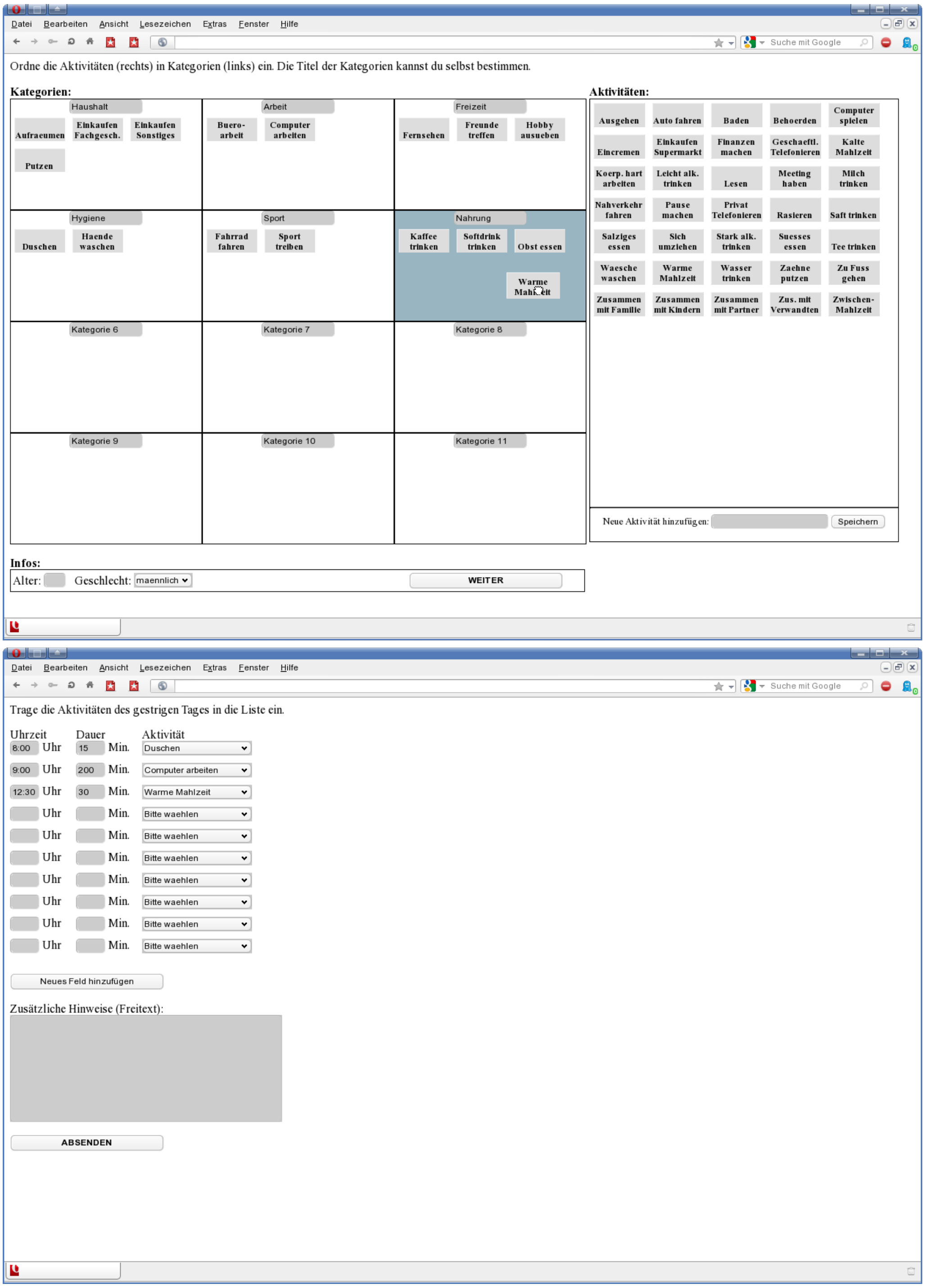


Figure 5.1.: Top: Screenshot of the preliminary questionnaire, the users can sort the activities listed on the right hand side into categories on the left hand side by dragging them with the mouse. Each category can be named. Missing activities can be added. Bottom: Second part of the preliminary questionnaire, the users are asked to define a typical day split up into main activities with their corresponding length.

5.2.1. Preliminary Questionnaire

People have very different and individual daily activities, which makes it hard to determine a set of activities which fits everyone. A basic set of available activities that has to be predefined can be used as a common composition for an efficient interface, and in this way help specify the current activities. The result of the preliminary questionnaire should cover the main activities which can be used later on in the study. Above and beyond that, in order to group activities these need to be put into predetermined categories.

In order to obtain these given sets of activities and categories, a preliminary questionnaire has been designed which presents and works on an already existing but not complete set of activities with the possibility of extending and adding missing items. The participants are asked to group the activities presented into up to twelve categories to which they can assign names. The result of this questionnaire is to aggregate a set of sorted activities and hopefully initially forgotten activities will be added by the users allowing the final set for the planned study to be as complete as possible.

The questionnaire was realized using an online website. It is split into two parts and begins with an introductory text which explains the task and gives some background information. The language of the content is German because the attendees are all German and it should also address people who do not speak English well enough to understand the questions and the procedure.

In first and main part the users have to sort the given predefined activities on the right into the category fields on the left (see figure 5.1 at the top). There are up to twelve categories fields available into which the activities may be sorted. The task for the users is to drag all activities with the mouse into category fields, and if activities are missing, they can be added to the list. Each category used by the participant has to be named. This name should reflect the activities and can be edited by the user. At the beginning the category names are numbered and the activity labels on the right sorted alphabetically. Activities can be moved afterwards, the order within a category does not matter. When the participant is done, the “next” button leads to the second part.

This step is not quite necessary for further evaluation but is used to get some general information about the personal perception of the users’ daily activities. The participant is asked to fill in all major activities of the previous day by selecting them from the drop down box of their previously created activities and specify a start time and the duration of the activity. Additionally, there is a free form field for entering notes about the questionnaire. The form is completed by this last step and the user gets to a “Thank You” final page.

5.2.2. Preliminary Questionnaire Results

The questionnaire was completed by 20 people. The mean age is 24.6 years, starting at age 16 up to age 39. More details on the distribution are depicted in figure 5.2. One participant did not provide any age information. There were 9 female (45%) and 11 male participants (55%).

The goal of the study is to find out what activities are important to people and how they can be categorized allowing a focus on exactly those activities. The first interesting finding is that nearly all participants grouped the activities into categories which are similar in

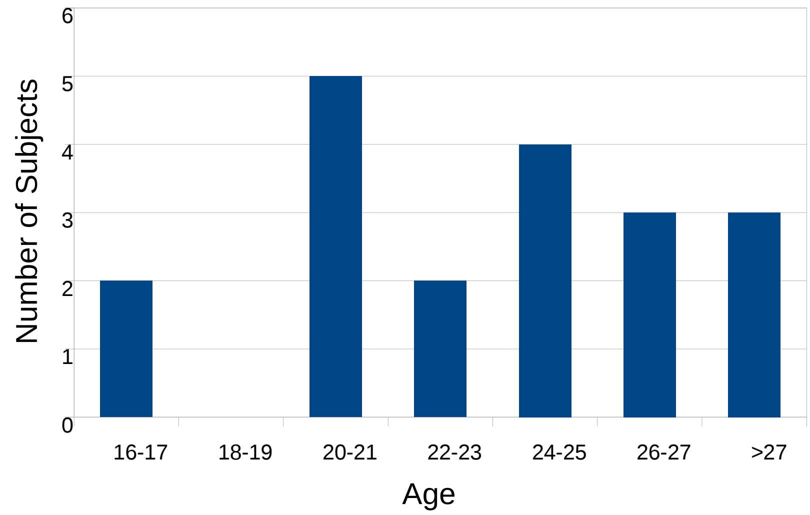


Figure 5.2.: Subjects of the preliminary study by age (one part. did not state the age)

content. The results lead to categories like “spare time” or “work.” There was one exception where a participant created time anchored groups such as categories like “morning” and “noon.” But as the majority worked with information driven categories, this is the first result of this questionnaire: Categories are dominantly created by information not by time. One fact that might have led to this result is that in the user interface each activity was only available once. This made it hard to create time categories as it would be necessary to fit one activity into multiple categories.

The participants were able to create their own category names, yet leading only to very minor differences in the labeling. For the evaluation, intuitive names were merged. From the result the following table shows any categories that occurred more than twice:

Category Name	**No. of Occurrences**
Eating and Drinking	19
Spare Time	17
Household	16
Hygiene	16
Transportation	15
Work	14
Social Contacts	14
Relax	6
Administration	5
Shopping	4
Miscellaneous	3

When the categories “Relax,” “Administration,” and “Shopping” are grouped into a general category “Miscellaneous,” there are eight categories. As information about “Eating and Drinking” can be very detailed and interesting, it is split into two individual categories: “Meals” and “Drinks.” It ends up with a total of nine categories, which can be grouped nicely in a 3×3 pattern. Nine categories allow a fast orientation for the users

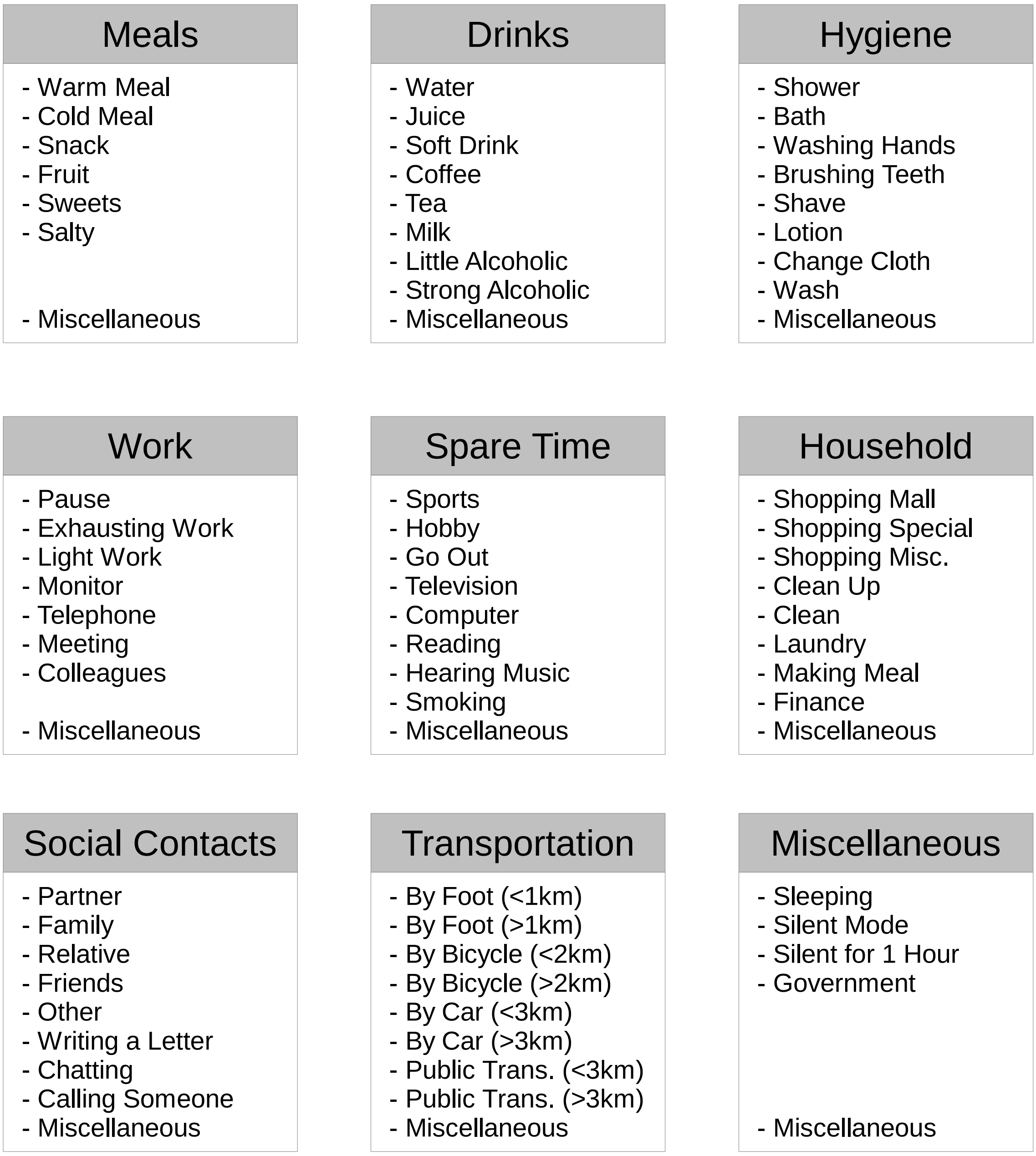


Figure 5.3.: Categories and activities resulting from the preliminary questionnaire. The organization in 3×3 categories enables a flat hierarchy and allows the users to access and add activities very quickly. Each category has a "Miscellaneous" activity for individual input if no activity in the category fits the need. This preserves at least the category of the entered activity.

and if each holds eight predefined activities and one free-form activity, there are 72 fixed choices altogether for the planned application. If an activity is not available or does not fit, the "Miscellaneous" activity can be chosen to obtain the category assignment. The complete resulting categories and activities are illustrated in figure 5.3.

The preliminary study delivered a set of categories that should fit the common needs of the users participating in the upcoming main study. The activities are hopefully complete at this point, but if there is still something missing, it can be sorted into the correct category manually. With a range of age and sex within the study, it should be ensured that the preselection is as good as common ground. The categories with activity sets provide a quick and efficient way to enter the current activity and can be easily implemented in software.

5.2.3. Conclusive Experiment Design

At this point, an application is needed for the experiment which runs on the selected phone and implements the previous decisions and the preliminary questionnaire results. The users have to be able to select from the predefined activities or enter a customized one if it does not fit the predefined ones. The application also has to provide an inactive state for privacy reasons and, not to forget, for sleeping time. When the participants are asleep the phone will not collect sensor data and, of course, will not interrupt the users until a previously selected timer restarts the process. As the phone will not be carried on the body, it is not necessary to collect data during this time, after all it is in an undefined condition. Typically, it will be charged and therefore lying on a table or somewhere else.

The users receive the mobile phone preloaded with the application. All they have to do is to turn on the phone. As soon as the phone is started, the application starts collecting data and automatically asks the users for labeling. A questionnaire before and after the study will complement the information about the users and their former experience with smartphones. Also, expectations and prejudices about a system like the one they have to use are asked in order to be able to compare the results before and after the study.

5.3. Software Implementation

The application was implemented using the Android SDK with compatibility to Android SDK version 5, which corresponds to Android 2.0 (*Eclair*). The application was developed to run and has been tested only on the target device HTC Desire. The application is called "ActivityTracker."

5.3.1. Experiment Software Operating Mode

Once the phone has been booted, the application is started automatically, the first time by the system itself. Each time the application starts, the first program task is to register the next alarm for three minutes. The alarm time is chosen so it triggers hourly (X:00, X:03, X:06, ..., X:57, (X+1):00, ... where X is the hour). This fixed timing makes the comparison of events easier at a later point. The next step is to keep the device awake with the display on for 25 seconds. This is the time needed to collect the 20 seconds of meaningful sensor

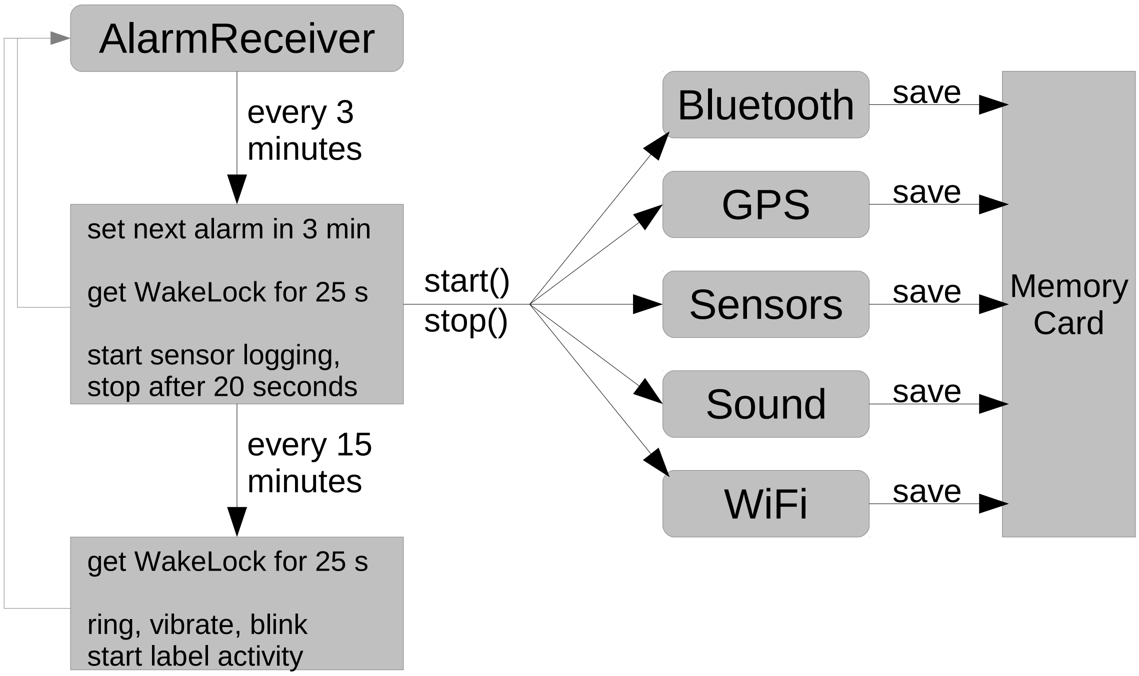


Figure 5.4.: Flowchart for the ActivityTracker application. The alarm is triggered every 3 minutes, every 15 minutes the user is asked for labeling input. Each sensor class saves its data directly to the memory card.

data and ensures some extra time for waking up the device and saving the data after the process has finished. The device has to be fully awake because some sensors do not deliver data when the display is off. Each sensor handles its system callbacks itself and saves the returned data on the memory card. After a collection time of 20 seconds the sensors are stopped. The complete application life cycle is illustrated in figure 5.4.

After collecting data there are two possible continuations: When the reason to start the collection process is a background alarm for sensor data collection, the application exits and waits for the next alarm by the system in order to continue. The user is asked to enter activity labels every 15 minutes; when this is the case, he has to enter the activity last performed since the previous entering. For that the main user interface is launched and the user is notified by a ringing tone, turning on of the vibrator, and lighting up of the built-in LEDs. The users may enter as many activities as they like and set the progress slider according to their current mood. The application can be closed when the user is done with the input.

The user will only see the application as the label input form. The background sensor collection is only recognizable by the back-lit screen during collection but will normally not be noticed by the user. The user can customize the notification type via the standard settings dialog of the phone. If the phone is generally muted, the application will not ring and the application uses the system-wide ring tone and volume for notification.

Aside from the normal activities there are two special ones: The first one is sleeping. When the user selects this activity, a pop-up asks for the time when to end this activity. In the sleep mode no data are recorded and the phone does not interrupt the user by alarms.

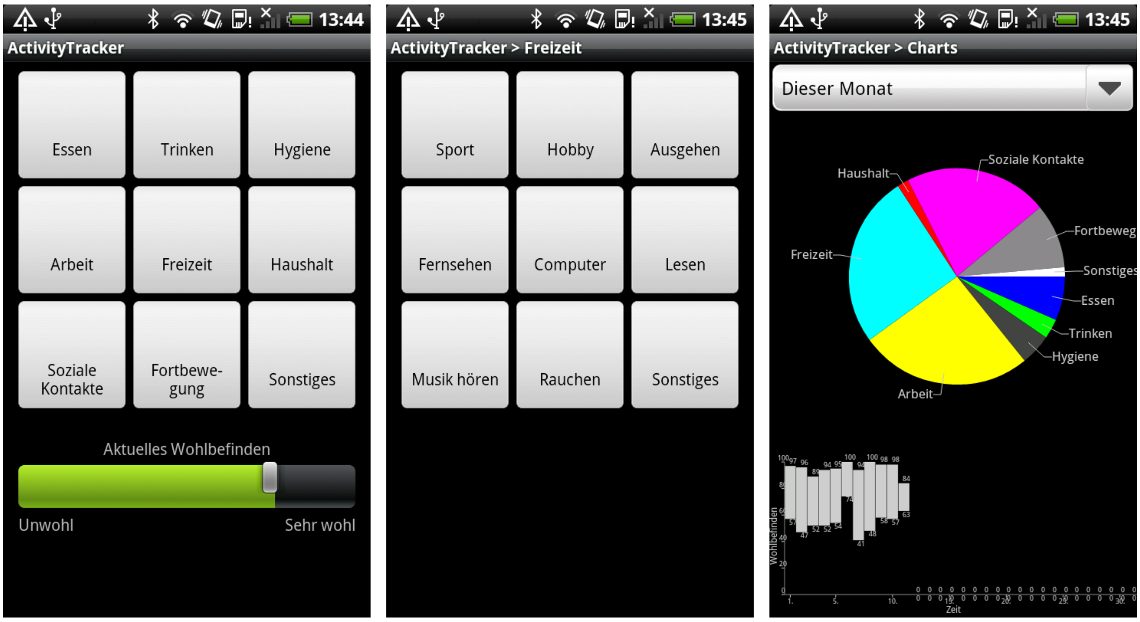


Figure 5.5.: Screenshots of the ActivityTracker user interface. Left: Main screen with categories and well-being slider below. Middle: List of activities in the category "spare time." Right: On-device summary of collected data. In study version all buttons were decorated with icons but had to be removed for copyright reasons. The interface is kept in German making it easier to use for the German participants.

The normal operating mode starts automatically when the selected time is reached. The same behavior applies for silent mode activities.

5.3.2. User Interface

The main user interface (cf. figure 5.5 on the left) is very simple and clear. It consists of nine large buttons, one for each category. The orientation on the screen is supported and made faster by providing the categories with small icons (removed in the screenshot for copyright reasons). At the bottom of the screen a slider is located to select the recent well-being value. The range is from "not well" on the left to "very well" on the right. The lack of discrete values is intended because the users should just enter an initial and rough value. After selecting a category, the user is directed to the activity selection screen (cf. figure 5.5 in the middle). A maximum of eight activities is shown and on the lower right there is always the free-form activity "Miscellaneous." The user is taken back to the main screen once he has clicked on the desired activity and the selection is saved automatically. This enables the fast entering of multiple activities.

Aside from the main screen and the activity selection screen, there is the possibility to review the entered data and to delete data. The review user interface has been implemented in a simple list view and can be accessed through the menu. There is also an overview which summarizes the activities in a pie chart (cf. figure 5.5 on the right) above a selectable time. Below the chart there is a daily overview of the well-being value range. This view gives the user a quick overview over his recorded daily activities.

The user cannot interfere with background sensor logging process. The interface is just for labeling the sensor data with activity labels and well-being values. The main screen pops up automatically when the system requests labeling input. It also appears when the application is started manually from the application list. The complete user interface is kept in German; all participants of the study were Germans. Not all of them were able to understand English and the system should be fast and easy to use.

5.3.3. Background Data Logging

While the user is queried in time intervals for labels, the software also operates in the background without notifying the user at all. When the background task is active, the application collects data from the smartphone sensors and writes them to the memory card. As the phone has to wake up from standby, the Android AlarmManager[2] API is used to schedule alarms. Before starting the actual data collection, the next alarm is set to ensure the next wake-up in case something goes wrong within this data collection trial.

Then the phone starts to query all available sensors for a time range of 20 seconds. This value is a compromise. On the one hand, it has to be battery efficient and, on the other hand, get as much sensor data as possible. Some sensors also need a certain amount of time until data is available, for example the GPS sensor. Here is a table of the collected data and the detailed logging format:

Sensor	**Logging Format**	**Frequency**
Bluetooth	TS;Device Name;Device mac address	variable
GPS	TS;Latitude;Longitude;Altitude;Speed; Bearing;Accuracy;Provider;Type	variable
Accelerometer	TS;X-Value;Y-Value;Z;Value Values in multiples m/s^2	~50 Hz
Light Sensor	TS;Value	only on change
Magnetometer	TS;Azimuth in degrees	~100 Hz
Proximity Sensor	TS;Value	only on change
Sound Sensor	TS;Max Amplitude	~5 Hz
WiFi	TS;List of Available Networks	variable

Each log entry starts with a time stamp (TS), followed by the individual sensor data. Within 20 seconds the sensors are queried as fast as possible, limited by the hardware and system implementation. For some sensors the frequency varies depending on the environment the phone is in. For Bluetooth and WiFi it depends on the number of devices / access points available. GPS is a special case as there are two types of position providers and two position types available:

- Last known position from network provider: This is the last known position which has been provided by the network provider. This position is based on the surrounding WiFi networks and can be outdated.
- Last known position from GPS sensor: This is the last known position from the GPS sensor. The position information can be outdated.

[2] `http://developer.android.com/reference/android/app/AlarmManager.html`

- Current position from network provider: This is a WiFi scan-based position estimation. The accuracy depends heavily on the environmental conditions.
- Current position from GPS sensor: This is the most accurate position from GPS. It takes some time to get a fix but then it provides the actual position approximately every second.

All types are logged so it can be decided which position estimation will be used later on. An individual log file is created for each sensor and the values are stored in CSV text file format with semicolons as separators. Figure 5.6 lists example data entries for each sensor.

5.3.4. Summary

An Android application was designed and built based on a general list of categories and activities that acquired with the preliminary questionnaire. Large buttons and icons enable the users to enter an activity with only two clicks on the screen allowing for fast and efficient data entry. The application is able to collect both sensor data and corresponding label information. Internal timers record data from all available sensors of the phone and save the results on the memory card. Customized activities can be entered manually and for sleep and silence two specialized modes are available which interrupt data recording and alarms. Overall, the result is a large set of meaningful, labeled sensor data which can be used for daily activity data evaluation.

5.4. Experimental Procedure

For the ongoing study each user receives detailed printed instructions for the experiment making the data comparable. The manual includes instructions how to operate the application and explains special cases. Summarized, this includes the following topics:

- Short description of what the application does in the background and what to do when the phone rings for activity input.
- The phone has to be charged over night to provide sufficient power for a complete day.
- The phone has to be worn in the front pocket and always in the same position to provide good sensor data.
- Short explanation on how the sleep and silent status works.
- Brief explanation on how to delete entered activities in case of accidentally/unwantedly entered items.
- Contact information in case of questions and problems.

The full instructions are attached (in German) in the appendix. Additionally, each user gets a personal introduction before the start of the study.

```
==================== Sensor Log File Examples ====================

Bluetooth:
  1297164975;Thomas Telefon;00:13:FD:93:B9:D0
  1297247762;Sara;54:92:BE:33:BA:AF
  1297249212;einwort (2);00:23:6C:A7:41:02

GPS:
  1297103761;52.025649;8.523377;0.0;0.0;0.0;60.0;network;lastknown;1
  1297103761;52.025671;8.523368;171.0;0.0;56.953125;32.0;gps;lastknown;1
  1297103763;52.025634;8.523350;0.0;0.0;0.0;60.0;network;listener;0
  1297103778;52.025687;8.523384;172.0;0.0;56.953125;32.0;gps;listener;0
  1297103779;52.025687;8.523384;172.0;0.0;56.953125;24.0;gps;listener;0
  1297103780;52.025687;8.523384;172.0;0.0;56.953125;24.0;gps;listener;0
  1297103781;52.025687;8.523384;172.0;0.0;56.953125;24.0;gps;listener;0
  1297103782;52.025634;8.52335;0.0;0.0;0.0;60.0;network;lastknown;2
  1297103782;52.025687;8.523384;172.0;0.0;56.953125;24.0;gps;lastknown;2

Accelerometer:
  1297074900;0.23154591;6.66035;6.3607025
  1297074900;0.23154591;6.701211;6.4424243
  1297074900;0.27240697;6.782933;6.4424243
  1297074900;0.27240697;6.8237944;6.4015636
  1297074900;0.313268;6.782933;6.4015636

Light Sensor:                    Magnetometer:
  1297156687;90.0                  1298028619;101.5
  1297156691;160.0                 1298028619;75.0
  1297156861;320.0                 1298028619;76.0

Proximity Sensor:                Sound Amplitude:
  1297612801;1.0                   1297080201;7077
  1297612819;1.0                   1297080201;5535
  1297612981;0.0                   1297080201;12956

WiFi:
  1297076763;start
  1297076763;SSID: ubi, BSSID: 00:23:04:36:0f:11, capabilities:
      [WPA2-EAP-CCMP], level: -88, frequency: 2412
  1297076763;SSID: eduroam, BSSID: 00:23:04:36:0f:12, capabilities:
      [WPA2-EAP-CCMP], level: -89, frequency: 2412
  1297076763;SSID: unibi, BSSID: 00:23:04:36:0f:10, capabilities:
      , level: -88, frequency: 2412
  1297076763;SSID: VPN/WEB, BSSID: 00:22:90:c4:64:a3, capabilities:
      , level: -89, frequency: 2412
  1297076763;SSID: guest, BSSID: 00:23:04:36:0f:14, capabilities:
      , level: -89, frequency: 2412
  1297076763;SSID: VPN/WEB, BSSID: 00:23:04:36:0f:13, capabilities:
      , level: -89, frequency: 2412
  1297076763;stop
```

Figure 5.6.: Examples of log file entries for each sensor.

5.4.1. Current Activity

Whenever the phone rings, the users have to take action and label the activities they did within the last 15 minutes. The application opens automatically and displays the activity input view showing the categories. The question to the users is always:

"What were your main activities in the last 15 minutes?"

The participants have to enter these activities using the category and activity buttons. The number of activities asked for should be self-limited to a maximum of three to get the most important. Fewer than three activities is fine, but there should always be at least one activity. When there are no fitting activities available, the users have the opportunity to enter a custom activity. In order to do that, a category has to be selected and then there is an activity "Miscellaneous" which opens a free text input form. There the users can enter a name for the activity. This input will be saved and preselected for the next time a custom activity of this category is needed.

When the labeling process is done, the application can be closed and the phone can be turned off again. The users do not have to hit any save button because the application generally works in such a way that any input is saved instantly. After another 15 minutes the application will again ask for activity label input.

5.4.2. Well-Being

Each time the users enter label data they are also asked to select their current "well-being" by setting the slider at the bottom of the application. This should just be a quickly determined self-evaluated value in order to get a rough understanding of the current mood of the users. There is no numbered value but instead just the slider, which ranges from unwell (slider to the left) to very well (slider on the right). Each movement of the slider is saved automatically when released.

5.5. General Experiment Questionnaire

Before and after the study questionnaires are handed out to the participants. After an introduction to the subject of the study, the first questionnaire is filled out by the participants. Aside from personal information, i.e. sex and age, the questionnaire consists of a few general questions about the expectations of the study and how the usage of the system will probably affect everyday life situations. At the end the users are asked to estimate the time they will most likely spend in certain categories during the study. The time is measured roughly in a percentage of the entire time. The complete study questionnaire handed out to the participants is available in German in the appendix.

After the users have completed the practical part of the study, they will receive a second questionnaire but this time a more extensive one. The first part is identical to the first questionnaire. This enables the analysis of changes during the study time. This is especially interesting for questions pertaining to how users thought the system would affect their daily activities and normal life. After the first part, questions about how well the

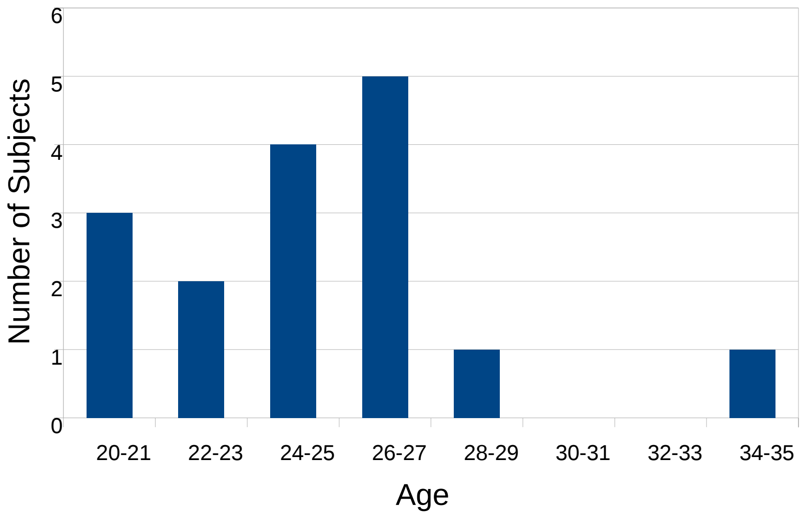


Figure 5.7.: Age distribution of the participants of the first study

system worked for the user and what effects a system like the targeted one could have on typical behaviors and habits were asked. Also missing categories and activities can be noted. It concludes with general questions as to whether the users worked with smartphones and Android before and about the functionality of the application. A free-form input is provided for general feedback. The entire post-study questionnaire is available in German in the appendix.

5.6. Experiment Conditions and Specifications

The main study was completed by sixteen people. The subjects were selected by contacting students and friends via e-mail or personally. Six of the subjects are female, which leads to a rate of 35%, and are ten male participants (64%). The mean age is 25 years. A more detailed break down is shown in figure 5.7. Each subject was rewarded €50.00 for participating in the study and completing the questionnaires. The participants each had to collect valid data for at least twelve full days in order to successfully finish the study and receive the payment. This requirement ensured a large and almost complete data set for later exploration.

5.6.1. Experiment Questionnaires Results

Only the direct questions of the questionnaires are discussed in this section. The detailed analysis of the activities in combination with the data which were collected during the study period are discussed in the next chapter. The following statements had to be rated in both questionnaires:

- Using the system will affect my daily operations. (affects d.a.)
- The interruptions by the system will disturb me. (disturbs d.a.)
- Using the system will affect me positively. (affects pos.)

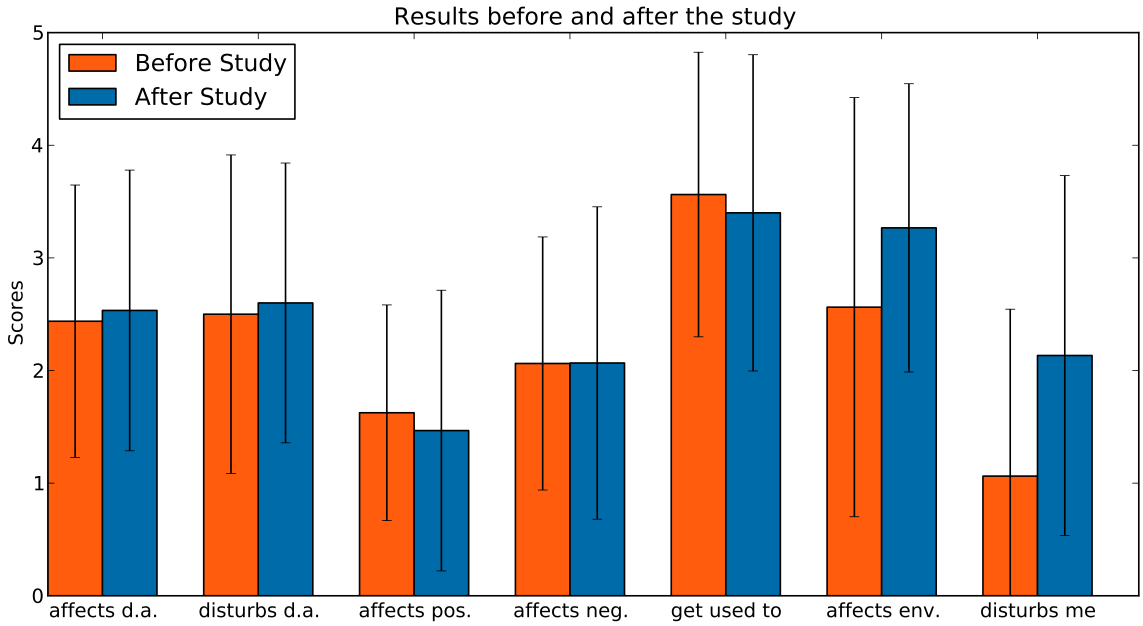


Figure 5.8.: Comparison of answers before and after the first study. A high score means that the subject highly agrees. The bars represent the mean values; the whiskers display the standard deviation.

- Using the system will affect me negatively. (affects neg.)
- I will get used to using the system. (get used to)
- Using the system will disturb my social environment. (affects env.)
- Using the system will disturb me in my social environment. (disturbs me)

The subjects were asked to answer on a scale from "does not apply at all" (score 0) to "completely applies" (score 5). The higher the value, the more the subjects agree to the statement. The first results of the questionnaire are displayed in figure 5.8. For each statement a double bar shows the mean value for all subjects. On the left hand side, the results of the pre-study questionnaire are displayed in red, and on the right hand side, the post-study results are displayed in blue. The standard deviation is plotted using whiskers for each individual bar.

The first two statements were answered nearly the same before and after the study with a mean centered value. The answers concerning the type of affect differ by about a half score point, here the negative aspect is slightly higher with a value of 2.0. As before, there is no notable difference between the results before and after the study. The highest values were reached with the statement whether the users will get used to the system. With a minimally lower score after the study, it is still very high at about 3.5 points. A tendency between the answers is notable for the last two statements. For the last statement the users answered one point higher and the bar rose from one to two on the score axis.

An interpretation of the results of the first two statements shows that the participants are unsure if using the system will disturb or even affect their normal daily activities. These results did not change after the study was over. When it comes to the statements as to

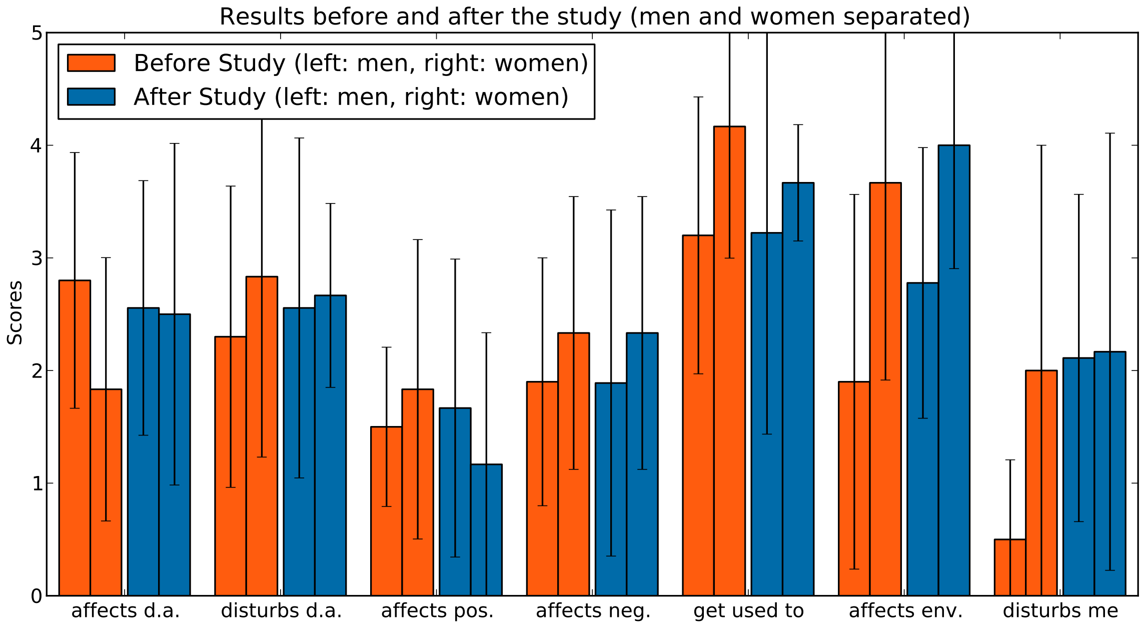


Figure 5.9.: Comparison of answers before and after the first study separated by sex. A high score means that the subject highly agrees. The bars represent the mean values; the whiskers display the standard deviation.

how the system affects the participants' daily life, it is subjectively more negatively than positively. Much more clear is the fact that the users think that they will get used to the system. Presumably because of prior knowledge of using applications on mobile phones or smartphones they assume that working with the system will get faster and easier in time. The standard deviation value of these statements shows that there were several users who fully or most likely agreed with the statement.

Now turning to the results which do show a noticeable difference between the two questionnaires before and after the study, the following observations can be made: Using the system will likely get the attention by the social environment like family and friends. With the instruction in mind to use the phone every 15 minutes, the users scored a middle value before the study. A high standard deviation indicates the wide spreading of the answers, which indicates that there were different types of users and environments. But after the study the higher score and the lower standard deviation for this item clearly shows that the users were wrong in their estimation. It looks as though using the system had a greater effect on their social environment than they originally thought. This effect is visible in the last statement as well. While the users did not think that using the system would disturb them during daily activities, the result after the study is a full score point higher. The users admitted that they underestimated the work and effort necessary to label the activities. At this point it would have been nice to know how the value would have changed continuously over time because this could have represented a break-even point when the users do not like to use the system anymore.

More interesting details become visible when the results of the questionnaires are split up by sex as seen in figure 5.9. Especially the responses to the last two questions differ

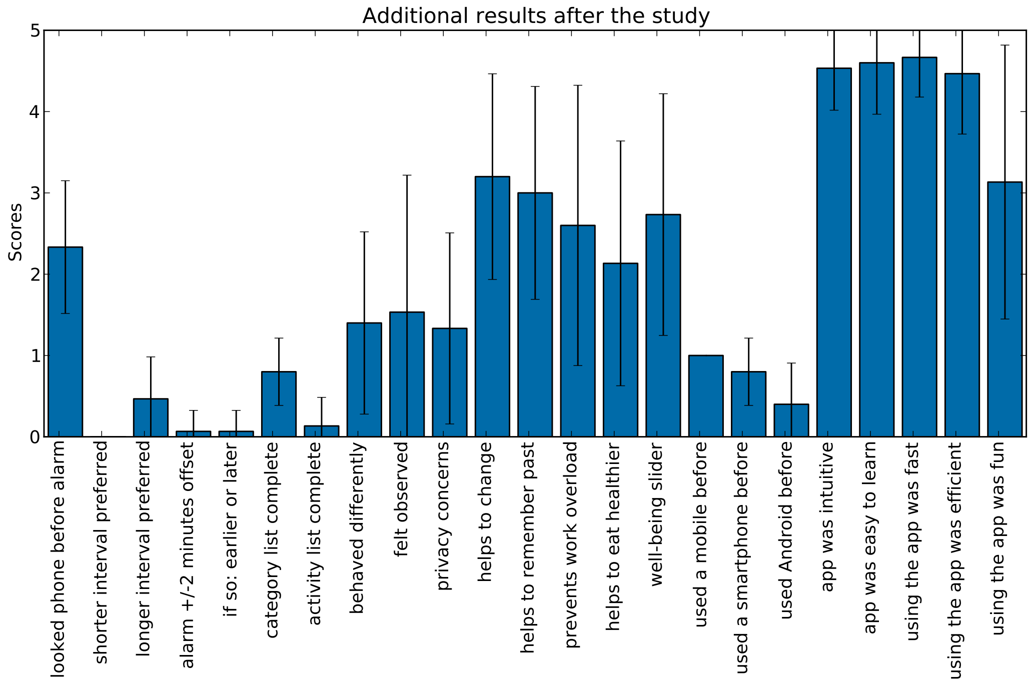


Figure 5.10.: Results of the additional statements of the questionnaire after the first study. A high score means that the subject highly agrees. The bars represent the mean values; the whiskers display the standard deviation.

between men and women. While the statement about the effects on the social environment do not show significant differences between before and after the study for each individual sex, the last statement does. The men answered with a score difference of 1.6 points significantly higher after they completed the study indicated by a t-test ($p = 0.01$).

The results of the additional statements of the questionnaire after the study are plotted in figure 5.10. Here is the complete list of statements asked from left to right. If not mentioned, the scale was the same as in the previous statements from 0-5. In binary statements the score was 1 for "yes" and 0 for "no."

- It occurred that I took a look at the mobile phone before the alarm ringed. (looked at phone before alarm, [never, seldom, sometimes, often, always])
- The interval should have been shorter (shorter interval preferred, [Yes/No])
- The interval should have been longer (longer interval preferred, [Yes/No])
- The interrupt time should have been 2 minutes earlier / later (alarm +/-2 minutes offset, [Yes/No])
- If so: earlier or later (if so: earlier or later [earlier, later])
- All categories have been available (category list complete, [Yes/No])
- All activities have been available (activity list complete, [Yes/No])
- I have behaved differently because of the system (behaved differently)
- I felt observed using the system (felt observed)

- I had privacy concerns using the system (privacy concerns)
- The application could help to change myself (helps to change)
- The application could help to remember better the past (helps to remind past)
- The application could help to prevent work overload (prevents work overload)
- The application could help to eat healthier (helps to eat healthier)
- The input of the well-being value was no problem (well-being slider)
- I have used a mobile phone before (used a mobile phone before, [Yes/No])
- I have used a smartphone before (used a smartphone before, [Yes/No])
- I have used Android before (used Android before, [Yes/No])
- The operation of the application was intuitive (app was intuitive)
- The operation of the application was easy to learn (app was easy to learn)
- The operation of the application was fast (using the app was fast)
- The operation of the application was efficient (using the app was efficient)
- The operation of the application was fun (using the app was fun)

The first five statements about the interval of the labeling process show that the users would have preferred a longer time between the labeling processes, no one would have liked a shorter period. This problem has been addressed before and for this study it was necessary to set this interval to the short value of 15 minutes in order to get good labels for the sensor data. For future versions this concern can be kept in mind allowing an increase in user acceptance. While the list of categories was mostly complete for most users, nearly everyone missed certain activities. So the pre-study results worked well for general categories, but, as expected, it was not able to cover every user's daily activities. Again, for future versions the application has to be able to learn new categories and activities and this way adapt to the individual user. Best practice would be to start with an empty set of labels and add new labels when needed. This allows a perfect fit for each user. However, for the first study this was no option as one goal was to be able to compare activity labels, which would not have been possible or only much more difficult with individually given labels.

According to the results of the statements about the behavior, the users did not have to change their normal daily activities very much while using the system. They did not feel observed by the system and had only few privacy concerns. This result is very important because such a system will only work when the users trust it completely. According to the statements as to whether such a system would help them in changing in any way, the users answered positively but had some concerns. Middle values were given for the statements about remembering the past better and work overload prevention.

Every participating user had used a mobile phone before, and 80% of them even a smartphone; 40% were familiar with Android phones. The application itself got best scores for intuition, ease of learning, speed and efficiency. Only the fun aspect was rated lower with a mean score of three.

5.6.2. Experiment Questionnaires Conclusion

The generally middle answer scores show some sort of uncertainness about whether a system like the used one will affect the daily activities in any way. The subjects at first did not believe that it has positive affects, but actually it does affect people in some way. Relatively confident answers were given for the statement whether they will get used to the system; and even after the study this indicated that the adaption to using such a system is very high. A general problem seems to be the social environment of the subjects, which is unfamiliar with the system and therefore wonders about the users who are using it so frequently. Very interesting is the fact that the men assigned themselves a very low score in the last statement before the study and reached the same higher value as the women after the study, who rated themselves the same both times. This leads to the suggestion that the assumed technology affine men had to correct their own estimate after the completion of the study.

It seems like the system was well accepted and that there are no critical problems using it. No participant discontinued the study before the scheduled end. The predefined values for categories and activities seem to have been well chosen; in the personal interview after the study the participants only noted that some activities were missing. On the whole, the selection was good. The system itself was accepted by the users, which could be seen easily as they had no problems using the application. Some user suggestions can be implemented in the next version. Overall, the participants gave a very good feedback and were very helpful for the ongoing development of further stages of the application and the system.

5.7. First Experiences and Lessons Learned

Aside from the collected data about sensor information and labels, this first study was about getting an idea of how users work with a daily activity diary system. The preliminary questionnaire results were the basic information on how to design an application which enables users to enter label data for categories and activities during the study. The final product was an Android application which collected the sensor data needed for the upcoming analysis in the following chapter. Therefore, it was an important step towards the next cycle of the system and in understanding daily activities. The application presented the users with a pre-selection of activities and collected sensor data automatically in the background.

Questionnaires before and after the study gave a deeper insight into the impact of the system on the users. Also problems such as missing activities and the time intervals were identified and can be avoided in the next cycle. The overall design of the application got a very positive feedback, which at this point permits the user interface design to be retained. Additionally, as this was the first larger user study, general knowledge about designing and organizing a study was gathered. It started with designing the questionnaires and led to trivial but very annoying problems such as failing memory cards during the study. The application itself did not once cause any problems.

Now the next step is to take a closer look at the collected sensor data and work with it. With the question in mind as to whether activities can be detected from sensor data, the data has to be examined and classification methods have to be tested.

6. Analysis and Evaluation of the Collected Daily Activity Data

A huge amount of data was collected during the study, which needs to be organized to get an overview and start the analysis of the daily activities. Over 300 MB of sensor data were collected for each participant, separated into sets by sensor, all tagged with timestamps allowing to connect matching data. At first the biggest challenge is to compute meaningful features from the raw sensor data (cf. research question 1). These features have to represent the data properly for a first visualization and analysis as well as for later evaluation and classification approaches using machine learning algorithms (cf. research question 2). These will be used to compact the data, which then allows a rough overview of the complete data sets. Connections and data relationships will become visible using suitable representations and help understand how to deal with the results.

After the analysis, the data have to be structured with the goal in mind to develop a classification system which is able to determine labels from sensor data. The selections of categories are evaluated and possible classification approaches for the computed features are examined. As the given task is no standard single-label classification problem, proper methods have to be found here to deal with the data. Aside from the classification algorithms, additional and suitable evaluation techniques are needed. By applying all these methods to the sensor data and features, the best classification approach has to be determined. This provides the basic functionality for the next cycle of the system in the following chapter, where the complete process of data collection, feature computation, and evaluation will be done directly on the phone.

6.1. Packing Raw Sensor Data and Computing Features

While the system is running, it collects data every three minutes for 20 seconds each. Depending on the sensor, data are aggregated with up to 100 Hz and saved on the memory card. In order to work with the sensor data, they have to be condensed. The sensor data can be combined during this time frame because the time of collecting data is considered one single time slot. Every 15 minutes the last five recordings are assigned to the labels given by the user. This may lead to some inconsistency and partially wrongly labeled features but a shorter interval of labeling was not reasonable for the sake of the participants. Therefore, the following considerations have to deal with this circumstance as labeling error.

28 features were defined, computed from the raw sensor readings, in order to obtain a reduced vector of tentatively meaningful indicators for the subsequent classification as follows: From the time value two features, $x_1 and x_2 (boolean)$ were derived. First x_1, whether the day is a working day (i.e. Monday to Friday), and, secondly, x_2 the time

of day since midnight in seconds. To preserve time distances within a day, the time was shifted forward by five hours. With this trick the time of a new day always falls into the sleeping time of the subjects. The beginning of the recording period is always used for the timestamp representing the data chunk.

The data of the triple-axis accelerometer is recorded during the 20 seconds measurement interval at a rate of approximately 50 Hz. For each axis of the accelerometer the following features were computed:

- Minimum and maximum value
- Mean value and standard deviation

For the amplitude data recorded from the built-in microphone, the same vector of features was calculated. For privacy and legal reasons the program does not store any recorded sound on the phone.

GPS and network data are available for localization features. Specifically, longitude, latitude, altitude, and speed as delivered by the GPS sensor were used as features. If more than one localization cue is available, the average position and speed is calculated and used. Also the mean compass bearing and its standard deviation over the 20 seconds measurement interval are used.

This is the complete list of computed features:

- (x_1) Working Day (Boolean)
- (x_2) Timestamp (time of day in seconds)
- Accelerometer X-Axis: (x_3) Min., (x_4) Max., (x_5) Average, (x_6) Std. Dev. value
- Accelerometer Y-Axis: (x_7) Min., (x_8) Max., (x_9) Average, (x_{10}) Std. Dev. value
- Accelerometer Z-Axis: (x_{11}) Min., (x_{12}) Max., (x_{13}) Average, (x_{14}) Std. Dev. value
- Amplitude: (x_{15}) Min., (x_{16}) Max., (x_{17}) Average, (x_{18}) Std. Deviation value
- Location by GPS: (x_{19}) Latitude, (x_{20}) Longitude, (x_{21}) Altitude, (x_{22}) Speed
- Location by WiFI / GSM: (x_{23}) Latitude, (x_{24}) Longitude, (x_{25}) Altitude, (x_{26}) Speed
- Compass direction: (x_{27}) Average, (x_{28}) Std. Deviation

This feature set or parts of it will be the basis for the upcoming methods and algorithms.

6.2. Data Overview and Structuring Daily Activity Data

The analysis was begun by selecting one participant and, prototypically, reviewing the data. In a first step, the feature data were plotted over the whole time of the experiment. Here the data were condensed again to time slots of one hour using the mean value in order to be able to plot the full range of time. The results can be seen in figure 6.1 and figure 6.2.

In figure 6.1 the participant's location was plotted. For each location provider (network and GPS) the longitude and latitude values are displayed. The absolute location was masked by providing relative position information only. The values are in degrees. One

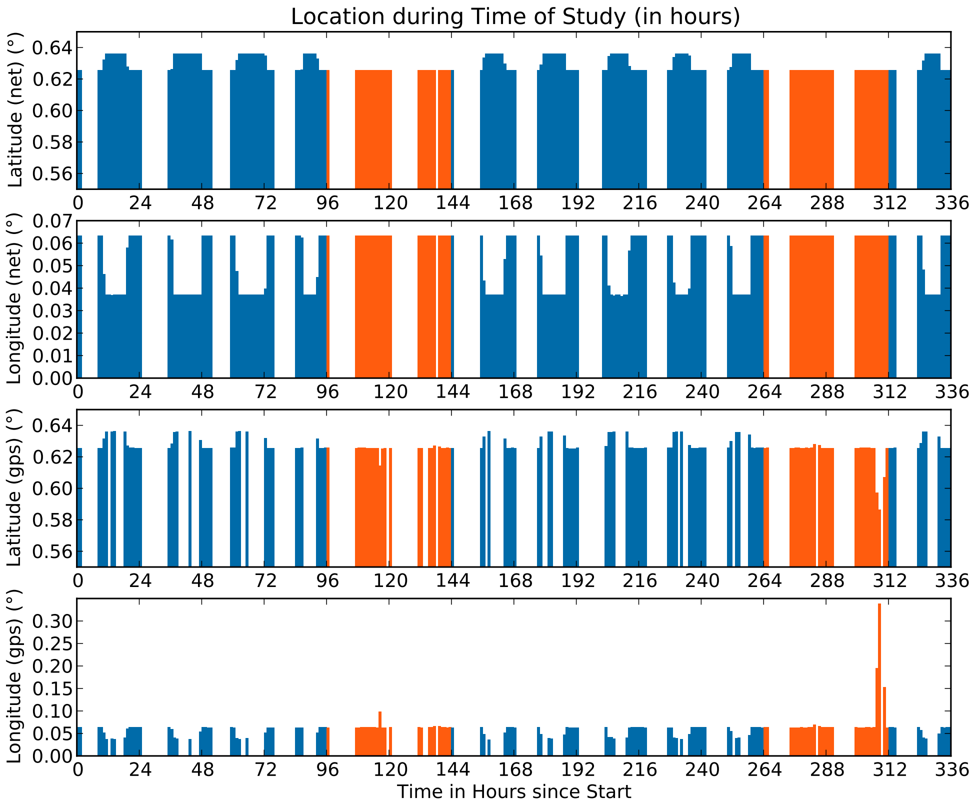


Figure 6.1.: GPS positions over the complete study time per hour; anonymized by removing exact location. Weekends (Sat, Sun) are displayed in red. Only available information was plotted; gaps in the day indicate missing sensor data.

degree longitude at the location of measurement correlates to a distance of about 69 km (one degree latitude always correlates to a distance of about 111 km, regardless of the position on the globe). Weekends are displayed in red, and the time since the start of the study beginning with the first full day is plotted on the x-axis. The displayed time is the real time of day, not the calculated, shifted time feature. The major gaps between the sensor values are sleeping times. At night the system did not capture any data.

Another point worth mentioning are the several small gaps in GPS sensor readings. As the GPS module needs to receive signals from the satellites, sometimes the device did not get a location fix and could not calculate the current location. This can be the case inside buildings or because of bad weather. It is also good to know that the network location provider returns the last known location if it is not possible to connect to the internet requesting the current location. This explains the relatively hard steps in the plot when the participants' devices did not have a cellular internet connection and had to rely on accessible WiFi networks.

As a regular pattern, two main locations can be determined. Clearly visible using the network data and confirmed by the GPS data, the participant is mainly located at two places, which also very much depends on the time of day. At daytime, a different location is preferred to the evening location, and again a different one for the night, i.e. sleep time. The two locations will very likely represent the home and work location of the participant.

At the end of the study at hour 310, one trip was made where the main area was left. The distance of this trip is around 20 km and lasted a few hours in the evening.

There are no easily recognizable patterns detectable when taking a first look at other selected features in figure 6.2. The mean amplitude value of the surrounding sound, as seen as in figure 6.2 in the first row, is taken directly from the Android SDK API. There is no documentation about the discrete meaning of this value available; it is only a rough linear estimation about the volume at the moment of recording. The most conspicuous pattern here is that on all work days there is a peak just before noon. This correlates to the time when the location was typically changed. There is one peak in the morning when relocating to the assumed work place and another one in the evening on the way home. These characteristics also show up during the trip at the end of the study. A conclusion might be that movement or transportation generally comes along with a high value in the sound amplitude readings. Over the course of the day the value is in the middle range and gets a little quieter in the evening.

The accelerometer values (by axis in rows two to four in figure 6.2) seem to partially correlate with the sound amplitude values. Body movement of the user continuously causes scratching noises from cloth because the phone is located in the front pocket. An interesting accelerometer axis here is the z-axis as it is normally at a high value. This is because of the orientation of the phone and earth's gravitation. A z-value of $9.81m/s^2$ indicates that the phone is lying flat on a surface, respectively a sitting position of the user. Small changes of the sitting positions lower the mean value. Generally, the user was sitting down for most of the day allowing the conclusion that it is an office job or something similar. Constant and low values in the evening could be activities like watching television. In general, these values are a good indicator for the level of activity and in the bodily position the user was in. For a more detailed analysis of activities a higher resolution over time and the combination of the remaining accelerometer features are crucial.

The built-in magnetometer returns the characteristics of earth's magnetic field. With the help of the accelerometer the actual bearing, like when using a traditional compass, was calculated. This value was plotted in the fifth row of figure 6.2. Combined with the standard deviation in the sixth row, it hints at the direction in which the user was going and to what degree the direction was altered during the measurement. The user will not change the orientation very much during low activity activities and when sitting at the desk or watching television; the rough orientation will always be the same. Very interesting here is the combination of a low standard deviation with the actual bearing because this behavior might indicate a specific activity very precisely in combination with other features.

Overall, the location is the strongest feature and will provide a very good indicator for activity categories like "work" or "spare time" in combination with the timestamp and the knowledge of whether the current day is a work day or not. Location changes are recognizable by changes in sound amplitude and accelerometer values and include hints about the type of transportation used. Inside buildings the location is too coarse to separate single activities but with the help of the magnetometer this might be possible.

The principal component analysis results, as seen in figure 6.3, reveal that about 90% of the variance is covered by the first five components. Therefore, many of the base features correlate and the set of categories can be mostly described with the first five components.

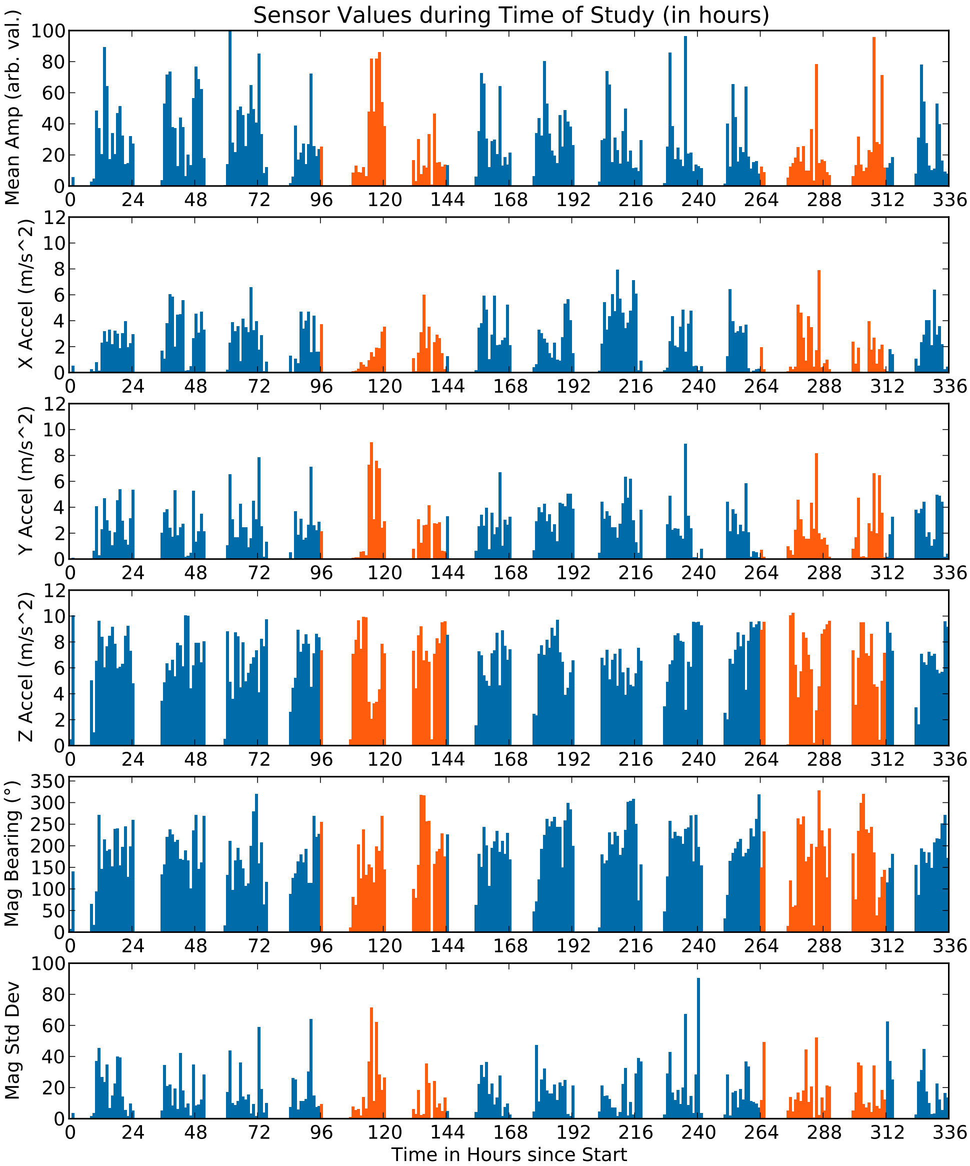


Figure 6.2.: Sensor values over the complete study time per hour, weekends (Sat, Sun) in red. From top to bottom: Mean amplitude of surrounding sound on a scale from 0-100 (arbitrary linear Android API value /2500), X/Y/Z mean accelerometer values, magnetometer mean bearing (degree) and its standard deviation.

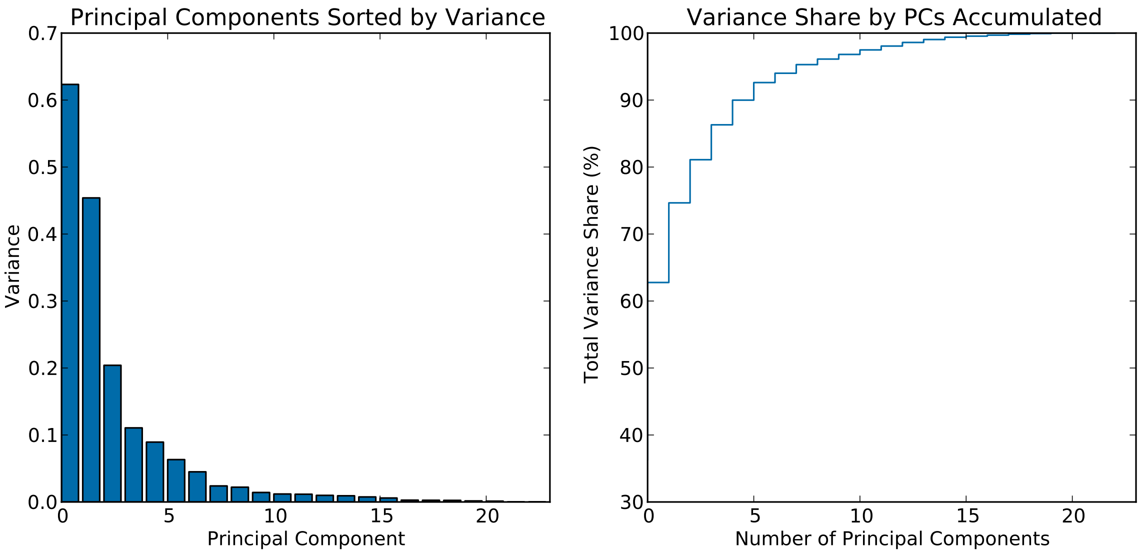


Figure 6.3.: Results of the principal component analysis. Left: Variances sorted by principal components. Right: Share of variance accumulated over principal components.

6.3. Visualization and Evaluation of Daily Activity Data

After reviewing the sensor and feature data, the next step is to connect these pieces of information with the recorded activity labels. The goal is to be able to classify activities by sensor data only; the values should indicate correlations between feature combinations and activity categories. For a higher level of abstraction and to reduce the complexity of the given task, from now on only the activity categories are used.

6.3.1. Data Visualization of One Full Day

Again the prototypical user is the data source. An overview of activities during a day is gained by looking at the combination of label and feature data as displayed in figure 6.4 for one full day. There are two iterations of the visualization available. The version at the bottom is the visually improved version, which will be the reference for the discussion.

Each of the ten activity rows from “Meals” to “Sleeping” in figure 6.4 shows a black square whenever the user selected the activity for the corresponding time slot. Multiple selections are visible by multiple squares in one column. After having started the day relatively late at about 9 a.m. with personal hygiene, the subject goes to work. In order to get to work a short distance of about 1 kilometer is covered. Until noon, work is the main activity followed by a meal at 12:30 p.m. Over the course of the day there are several contacts with other people, probably coworkers. Work ends at about 6 p.m. Now the evening starts dominated by spare time. Noticeable here is the contact to other persons that could be the user’s family. The day ends (not visible on the chart) at 1 a.m. Generally, there are always longer periods of the same activity category until the next larger time frame begins. Aside from “sleeping,” these categories are “work” and, in the evening, “spare time” often combined with “social contacts.”

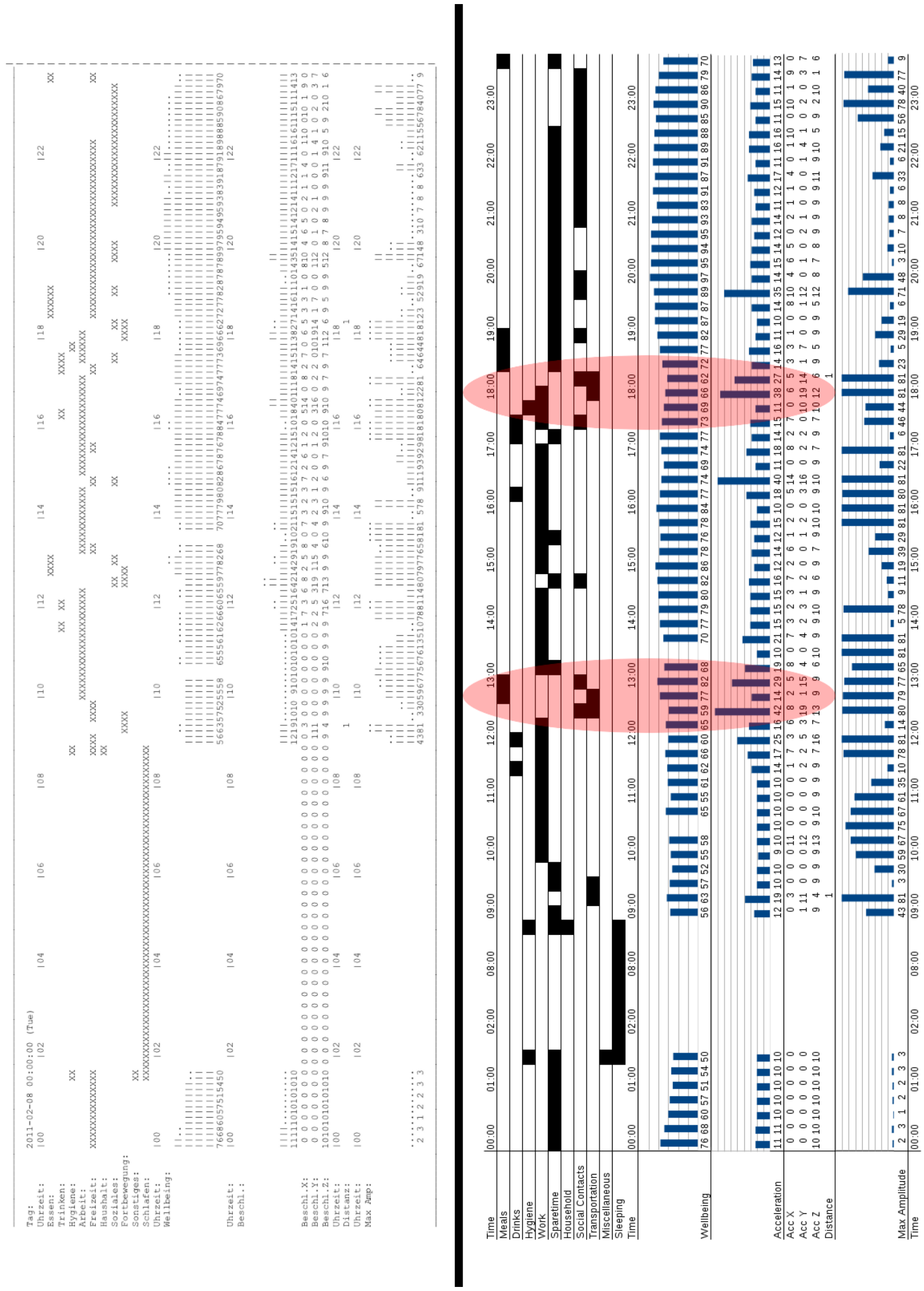


Figure 6.4.: Visualization of daily activity categories and sensor data for one day. Top: Original visualization, ASCII art; Bottom: new, graphically improved visualization, sleeping time collapsed. Rows from top to bottom: 10 categories (black square displayed if selected), plotted well-being data, acceleration sensor data, traveled distance in km, sound maximum amplitude.

The well-being value was plotted below the activities. The scale ranges from 0 to 100, the value is subjective and will not influence the further investigations. It was recorded merely for statistical analysis. The acceleration sensor plot combines all three axes measured in multiples of $1/10 * g$ where $g = 9.81m/s^2$. Below this is the detailed value for each axis. The distance row shows the distance covered since the last 15-minute time slot. For each time slot the center has been averaged and the distance is measured to the last time slot center. Only distances greater than 1 kilometer are visible. The average sound level for a time slot is plotted in the last row. The value range is scaled to a range from 0 to 100.

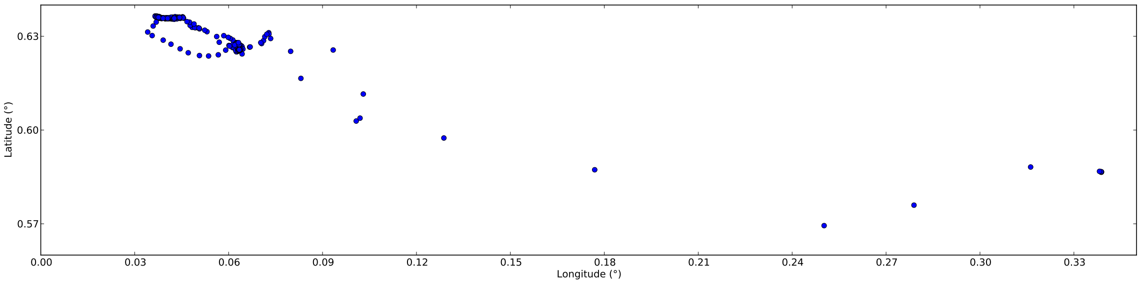
There are peaks every time the user moves from one place to another, as assumed before, by looking at the accelerometers. Twice distances larger than 1 kilometer can be seen, which corresponds to the way to work and back home. These should be good additional indicators for transportation. Also interesting is the sound amplitude which is relatively high all through the day. In the evening it gets lower, indicating a silent activity such as reading a book or watching television. When an interaction with other people starts, it goes up again.

There are two red ellipses drawn marking interesting changes of activities during the day. The first time slot is about noon, when the participant had a meal. It is clearly a break during the day and the sound amplitude sensor readings already give a hint, as they were exceptionally high at this time. At the end of the day, marked with the second red ellipsis on the right, the user finished work and went home. After this event, the accelerometer values are relatively low, indicating that there was not much physical activity. The labels “spare time” and “social contacts” indicate conversations with family members while watching television.

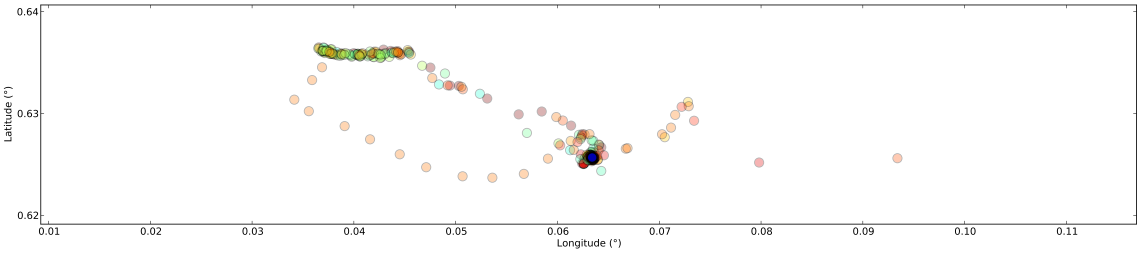
6.3.2. In-Depth Evaluation of Location Data

When taking a detailed look at the location data of the participant, it becomes obvious that there is a main action area. In figure 6.5a the GPS location data are plotted showing this area at the top left. In order to provide anonymity, the values are masked again by translating the real location points to a new coordinate system. The distances within the plot are kept. Each point displays the mean location of one 20 second recording, which was done every three minutes. The largest cluster of points is located at the values 0.06 longitude and 0.625 latitude. From here there were several movements east, but most points leading west, probably the work location with the daily way in the morning and evening. The path in the south indicates an alternative to the main route. As mentioned before, the user took one longer trip at the end of the study, which is now visible as a trail leaving the main area to the right. The trail leads to the east where the user stayed for some time and then went back. The distances between the points on the trail are a cue for a fast method of transportation like the train or a car.

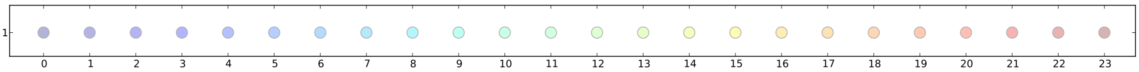
A close-up view of the main area is shown in figure 6.5b. This time each point is color-coded by the time of recording. The legend is available below in figure 6.5c. Revisiting the largest cluster, it now becomes visible that the points were mainly recorded in the evening and morning hours, which is a strong indicator for the place where the person is living. The same logic applies to the area at the top left. There the user typically spends his, probably at work. Here the points vary a little bit, which can be caused by jitter, bad GPS signal, or simply moving around in the work area. Also GPS location data are not



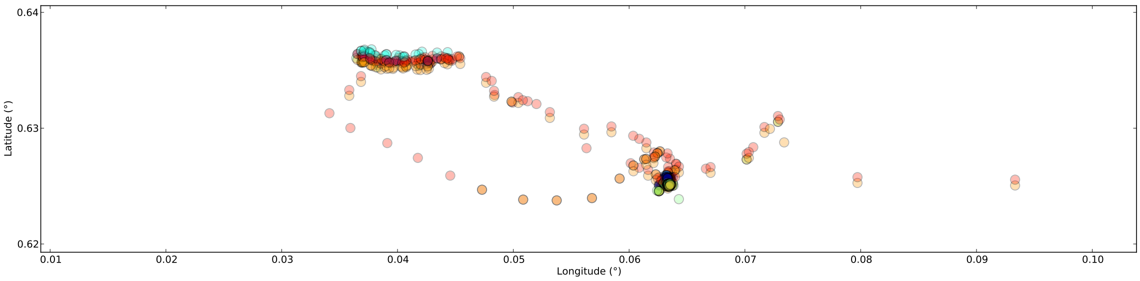
(a) GPS track overview of one participant. Generally, the main locations are in the top left area but there is one trail leaving the main area.



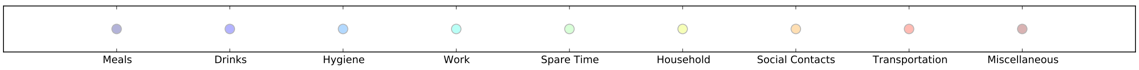
(b) Detailed view of the upper left part of the track. The hour of the day is color-coded, it is clearly visible that the locations at the top are typically visited at a different time than the locations at the center.



(c) Color code for each hour of the day.



(d) Detailed view of the upper left part of the track. The recorded activity is color-coded, it can be seen that the locations typically are connected to activities.



(e) Color code for each activity.

Figure 6.5.: GPS track of a selected participant, overview and detailed view. The devices recorded the data for 20 seconds every 3 minutes. Recorded locations are summarized and visualized by a circle. Values are latitude (y-axis) and longitude (x-axis) in degrees, anonymized by removing exact location, distances unchanged.

always available when inside a building. This could be the reason as to why there is no dense cluster at the office location. With the time coded view it is also discernable that the south trail was recorded on the way home from work in the evening and, as known before, the trip leaving the home area was in the evening as indicated by the red circles in the east. Altogether, the days of the participant strongly depend on the time of day and refer to structured daily routines.

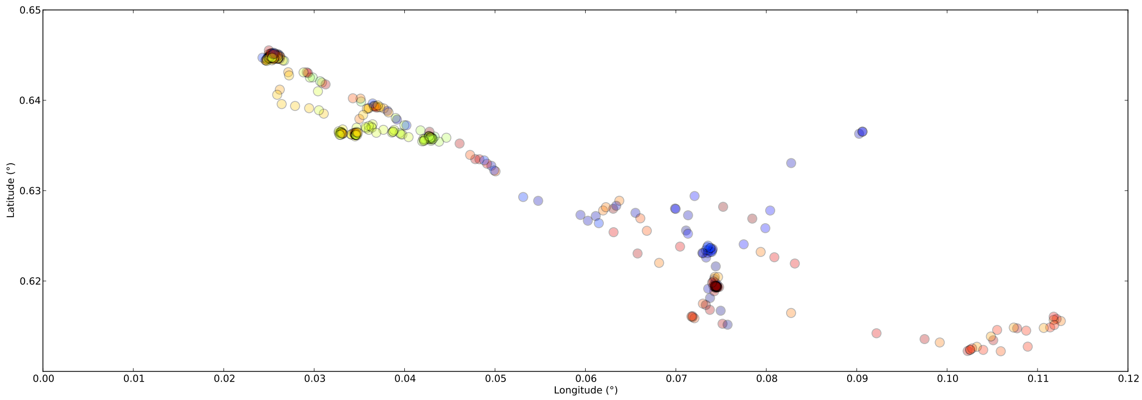
The color-coding of the GPS locations by activity category as in figure 6.5d (legend below in figure 6.5e) reveals that activities are often connected to specific places. For a clearer picture a little noise has been added to each category by minimally translating the points vertically. "Household" and "Spare time" are mainly recorded at the home location. The same applies for "Work," which is condensed and available only at the top left. "Meals" and "Drinks" are only visible at home but not finding these activities during work could be because of a lack of a valid GPS signal at the dining place. "Social Contacts" were recorded at both places and even sometimes during "Transportation." As this category includes moving on foot as well as going by bus or train, the concentration at the work place can be explained by short ways during work on foot. For "Social Contacts" there is no obvious specific location so the participant seems to be with other people for the most of the time during the study.

For reference the locations, color-coded by time and activity category, were plotted for another participant in figure 6.6. The area of this user is about two times larger than of the first one. Color-coded by time (figure 6.6a), there is again one location recognizable for the night (0.073° / 0.622°) in the center and one for the day on the left (0.035° / 0.635°). But in comparison to the other user, there is a third region in the top left corner visited in afternoon and evening, and another one to the right of the day area. In the bottom right several late evening stays can be seen. The color-coding by activity reveals that "Work" is concentrated at the day place and "Social Contacts" and "Household" are concentrated at the night location. At the top left there seem to be some friends activity as there is an aggregation of "Spare Time" and "Social Contacts" points. Meals were mainly taken near the work place and the place to the right of it. The short distances between "Transportation" spots refer to going on foot and again "Social Contacts" are spread all over the map alongside all other categories. Longer distances were covered by faster transportation methods.

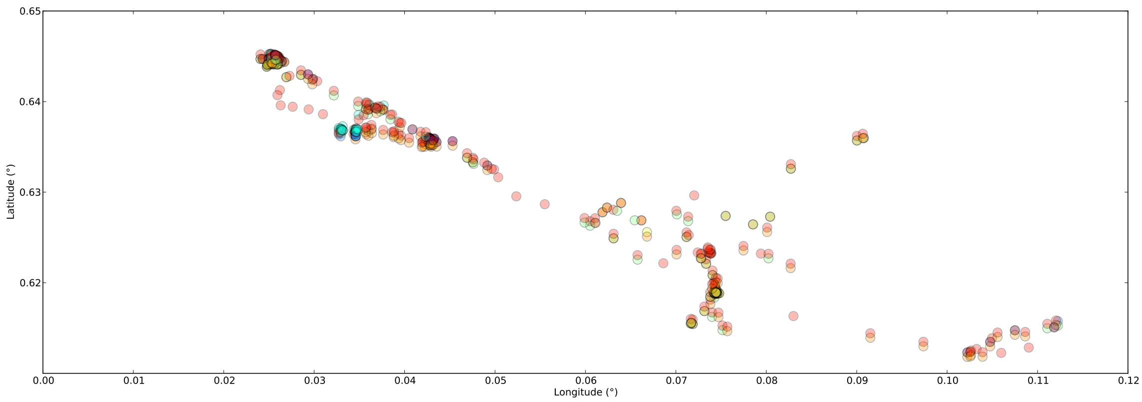
In comparison to the first participant selected, the second one visited more than two places regularly. Although the main area is larger, the second user did not leave it once during the course of the study. All over the map there are small concentrations of points which probably indicate casual meetings with friends in the city. Overall, the day is not as strictly structured as the one of the first participant but again for some activities the location information should be a strong clue for the classification.

6.3.3. Evaluation of Other Feature Data

Aside from the location provider, three other sensors might provide information about the current activity. For an analysis, a quick look through the data recorded of the sound amplitude, accelerometer, and magnetometer sensors will give more information about the characteristics for each activity label. Therefore, for all three sensors the mean value for the corresponding category over the time of the study were plotted along with the



(a) Time color-coded visualization of another participant.



(b) Activity color-coded visualization of another participant.

Figure 6.6.: GPS track of another participant with a relatively small action radius during the study time. Color codes are the same as in figure 6.5.

standard deviation in figures 6.7, 6.8 and 6.9.

Beginning with the amplitude of the surrounding sound, the combined mean amplitude values over the complete study time were plotted in figure 6.7. The value on the y-axis is again the previously described arbitrary value resulting from the Android API call. The most eye-catching bar is the one of the "Transportation" category, which displays the highest value. This corresponds to the earlier findings that when the user changes the location, the sound amplitude value is always very high. Above and beyond that, during "Household" activity there is generally a high volume level which may be connected with the use of a vacuum cleaner. "Meals" and "Social Contacts" are showing similar values, which might be due to the fact the meals were eaten together with other people. The lowest score after "Miscellaneous" is "Hygiene." Overall, the sound amplitude value is no key feature for the most categories but at least for "Transportation" it might be a good hint. For all other situations it can be a good addition together with the remaining sound amplitude features, but this always depends on the individual case.

Looking at the accelerometer values in figure 6.8, the main conspicuousness can again be seen in the "Transportation" bars. Here the value of the highest axis changes from the z-axis to the y-axis. This is because transportation mainly takes place while standing

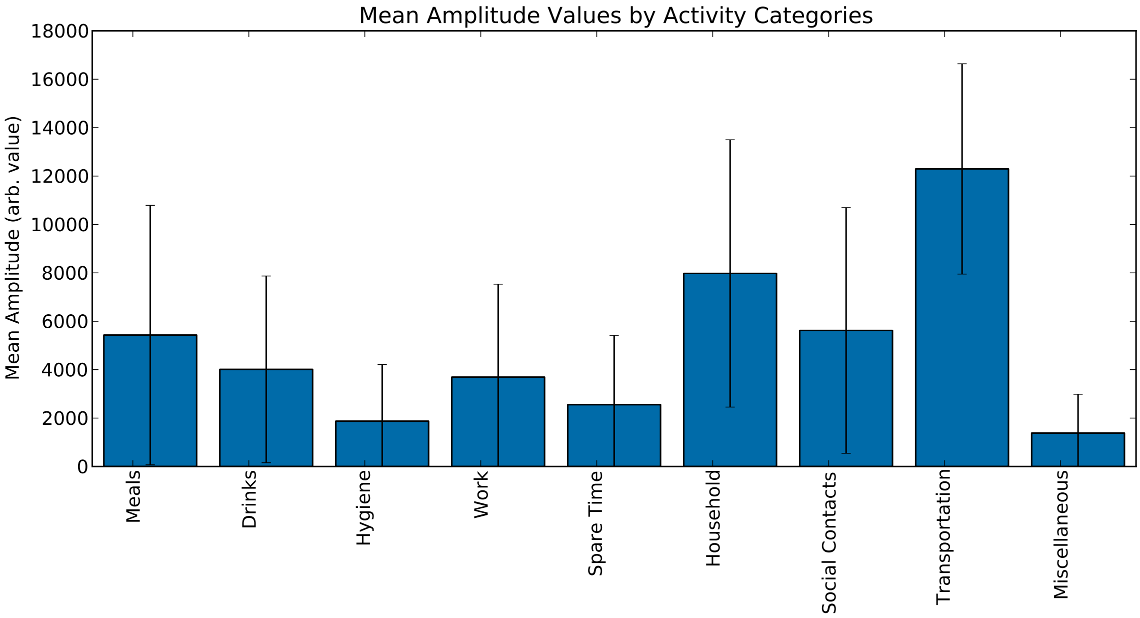


Figure 6.7.: Mean amplitude values of surrounding sound plotted over the whole time of the study by activity categories with standard deviation.

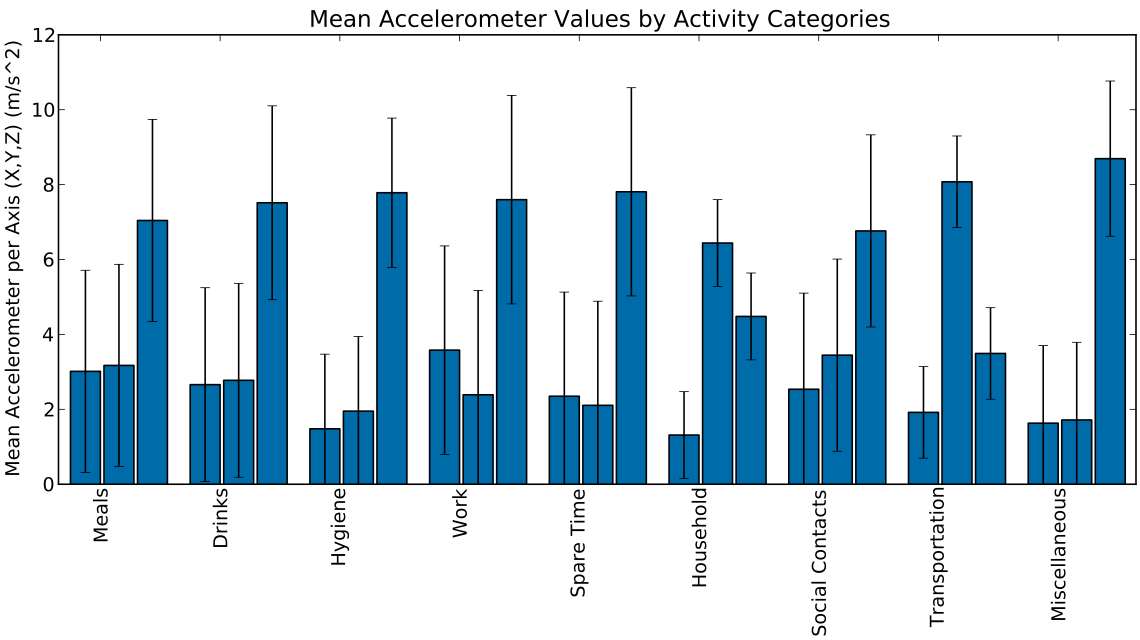


Figure 6.8.: Mean accelerometer values plotted over the whole study time by activity categories with standard deviation. For each category the three axes (X,Y,Z from left to right) were grouped.

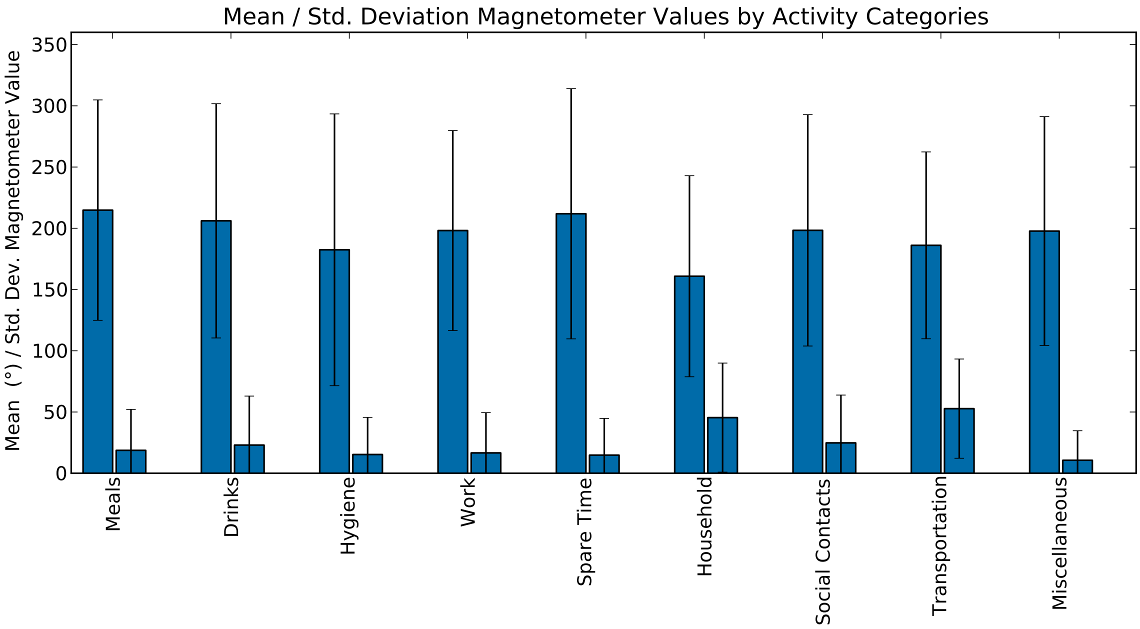


Figure 6.9.: Mean / standard deviation magnetometer values plotted over the whole study time by activity categories with standard deviation.

and the phone no longer is in a lying position. For all other categories it is again hard to determine any characteristics and a more detailed analysis of each individual case is necessary.

The mean compass readings were assumed to give some information about the preferred direction the user is looking at during an activity. Reviewing the recorded data as plotted in figure 6.9 for every category, the mean direction is about 200 degrees with a very high standard deviation. The standard deviation during the recording time is plotted right to the mean value and contains information as to how much the direction has varied. The low value for most categories shows that the users did not change the orientation during the measurements with the exception of "Household" and "Transportation." Again, these findings coincide with the previous results. Overall, there seems not to be a general orientation for a specific category.

6.3.4. Activity Category Combinations and Transitions

Putting the computed feature data aside and taking a look at the label data, only two aspects are interesting. First, the number of concurrent activities has been calculated for the two participants in table 6.1 and table 6.2. The users were instructed to enter at least one and up to three labels whenever the phone was ringing. The second participant did not always respect these instructions and limits, because the maximum of recorded labels is six. The mean value of activities during the study time was 1.5 and 1.8.

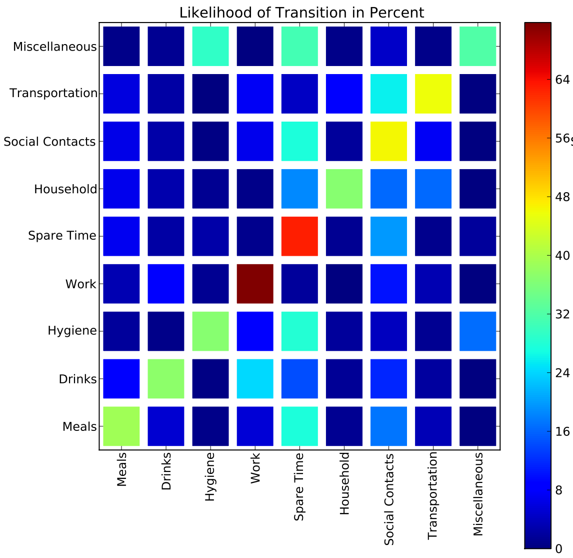
The first participant is having meals and drinks mostly together with other activities. This could be reciprocal behavior, but for over 30% of the time at least a second category was added. During work there were no distractions through other things. Only in 30% of the

Table 6.1.: Number of combinations of each category with others of the **first participant**. Labels on the left, single occurrences and combinations with other categories in percent. Minimum: 1 category, maximum: 4 categories, mean: 1.5.

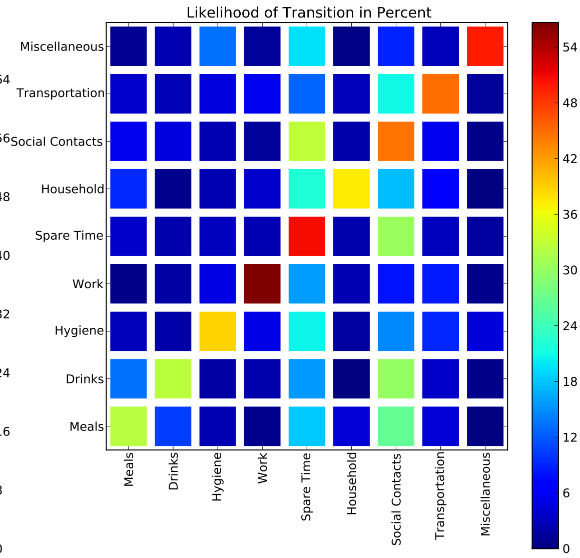
	Single	**One Other**	**Two Other**	**>Two**
Meals	2.2 %	67.0 %	30.8 %	0.0 %
Drinks	1.8 %	61.9 %	32.7 %	3.6 %
Hygiene	7.7 %	42.3 %	50.0 %	0.0 %
Work	69.1 %	26.6 %	3.9 %	0.4 %
Spare Time	52.4 %	39.1 %	8.3 %	0.2 %
Household	7.7 %	50.0 %	38.5 %	3.8 %
Social Contacts	14.1 %	69.0 %	16.1 %	0.8 %
Transportation	25.0 %	56.6 %	17.1 %	1.3 %
Miscellaneous	0.0 %	21.4 %	78.6 %	0.0 %

Table 6.2.: Number of combinations of each category with others of the **second participant**. Labels on the left, single occurrences and combinations with other categories in percent. Minimum: 1 category, maximum: 6 categories, mean: 1.8.

	Single	**One Other**	**Two Other**	**>Two**
Meals	0.0 %	31.0 %	55.2 %	13.8 %
Drinks	0.0 %	38.3 %	48.9 %	12.8 %
Hygiene	21.7 %	43.3 %	26.7 %	8.3 %
Work	47.5 %	40.0 %	11.3 %	1.2 %
Spare Time	24.1 %	62.0 %	11.8 %	2.1 %
Household	15.8 %	42.1 %	36.8 %	5.3 %
Social Contacts	6.6 %	73.5 %	17.1 %	2.8 %
Transportation	24.6 %	55.8 %	15.7 %	3.9 %
Miscellaneous	45.2 %	32.3 %	16.1 %	6.4 %



(a) Transitions from one activity category to another of one selected participant.



(b) Transitions from one activity category to another of another selected participant.

Figure 6.10.: Transitions of activity categories of two participants. The transition probability is always from the on at the y-axis to the one of the x-axis. Note the slightly different scale of the second participant.

time were there other, combined entries. While “Spare Time” accounts for 50% alone, the ”Household“ category rarely does. Not surprisingly, ”Social Contacts“ accounts for one part of a combined category in over 85%. The second participant shows more combinations in general, and the number of combined activities is higher. Especially ”Social Contacts“ is only in 25% of the cases a single category.

Also worth analyzing are the transitions between categories. Visualized in a transition matrix the results for both participants are shown in figure 6.10a and 6.10b. The matrix has to be read from the y-axis to the x-axis. So for the transition value from ”Social Contacts“ to ”Spare Time“ the crossing of the third row with the fifth column holds the corresponding value. The value itself is in percent color-coded using the legend on the right. Note that the legends for the figures are slightly different.

The high diagonal values show a high probability that the same category as before was selected once again. More interesting are the changes to other categories. Here the most outstanding values are in the columns of ”Spare Time“ and ”Social Contacts,“ columns 5 and 7. This means that there is a good chance that the following selection will be one of these.

6.3.5. Conclusion

In this section the recorded data for a prototypical participant was visualized, analyzed, and evaluated. In order to concentrate the huge amount of data, a set of features was constructed. Plots of these features over the complete study time revealed first pieces of information about the situations and locations of the user. The principal component analysis showed that the dimensions of the feature vector can be condensed to five dimensions while keeping 90% of the variance. More information about the daily activity is available by going into detail, reviewing one full day with first observable correlations between feature data and activity categories. An in-depth inspection of the location data recorded by the phone's GPS gave a good idea of the daily movements of the participant. Main location spots exist and inspecting the data together with time and activity information, assignations for locations can be determined. A second participant shows different movement behavior, but again it is possible to assign some activity categories to specific locations. The rest of the features also showed some characteristics, but not as concisely as the location data leaving it as a main indicator for later classification.

In summary, it seems to depend on the category how well actually the approach of classification by the computed features works. For some categories it seems to be easy to recognize these features, but for others it seems be relatively much harder. The correlations in-between the label data were helpful in the correct classification and should for this reason alone be included in the method. Also, it is very important to choose a method that is able to deal with more than one activity label at a time. This scenario was frequently observed in the participants' behavior. The next step now is to find a suitable algorithm to solve these problems and test it against all participants.

6.4. Classification of Daily Activities Using Sensor Data

The recorded data consists of a set of features recorded every three minutes. For each feature set a vector of corresponding labels given for 15-minute timeslots is available. The label data is always applied to all five chronologically fitting feature sets. The distinctive feature of these label sets is that more than one label can be active at one time. In order to conserve the connections of label combinations, it is crucial to work on these combinations as a whole. Examining each label individually would destroy this connectivity. Therefore, traditional single label binary classification algorithms cannot be used. To solve the problem presented, multi-label methods must be applied.

6.4.1. Classification Methods for Multi-Label Problems

In general, there are two ways to deal with multi-label problems. The first is to create new, specialized multi-label classification algorithms that are able to deal with multiple labels. The second way is to transform the original problem into a set of binary classification problem. After the transformation the traditional methods can be applied. Both types will be tested and compared by selected available, well-proven methods.

The Multi-Label k Nearest Neighbor (MLkNN) approach is a true multi-label method. Derived from the traditional k-nearest neighbor approach like the one presented by Cover and Hart [23], the algorithm was adapted to work for multi-label problems. The MLkNN method used was developed by Zhang and Zhou [95] and performed very well compared to other methods with their test sets.

The second method is to generally use a label powerset to transform the problem into a single label problem. In order to simplify the result the *RAndom k labELset* (RAkEL)

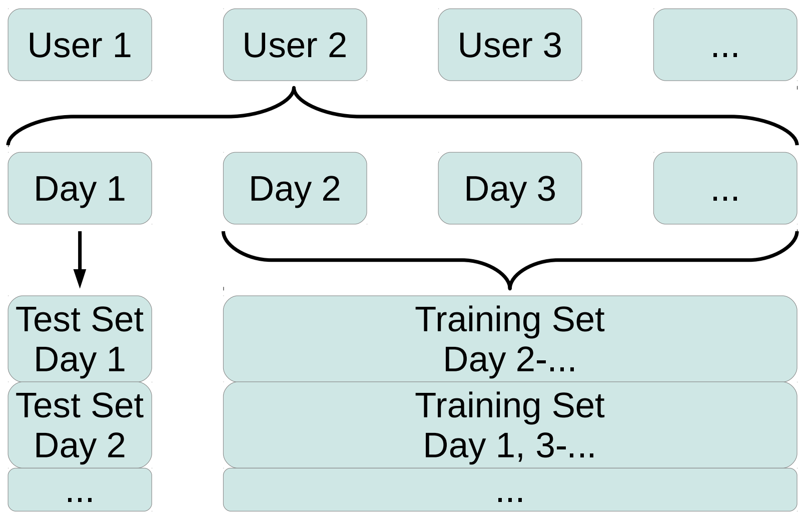


Figure 6.11.: Leave-one-out method to create test and training data sets. The data of one full day was picked for each user as a test set for the remaining data. This was done for all recorded days. The results are as many test/training sets as days available.

algorithm as proposed by Tsoumakas et al. [86] was applied. The label powerset method creates new single label classes by using the existing multi-label sets. It generates a large amount of new classes which include every combination of available label combinations of randomly chosen examples. By using this method, the correlations within label combinations are preserved. The RAkEL method breaks the set of labels down into smaller label sets and then computes a label powerset for each one. This reduces the number of possible combinations by preserving comparable classification results. The resulting ensemble of label sets can be used with a single label classifier. Here, a pruned decision tree is used. The implementation is called J48 which is a C4.5 decision tree as described by Quinlan [69].

The open-source tool Weka was used for classification. Weka (Waikato Environment for Knowledge Analysis [38]) is an open source collection of visualization tools and algorithms for data analysis and classification. It is written in Java and developed at the University of Waikato. It allows the testing of collected and preprocessed data quickly with several machine learning algorithms. In order to solve multi-label problems with Weka, an extension called Mulan [85] is available, which is is also open source. Mulan offers a variety of multi-label problem solving algorithms.

Sets of test and training data were derived from the original data in order to test the classification algorithms. This was done for each user individually and followed the procedure as seen in figure 6.11. For each recorded day, this day is extracted from the complete data set and forms the test set. The remaining data are used to train the classifier. So there are as many sets available as days of data recorded. The final classification result is the mean value of all data sets per person. The extraction of one full day has advantages against other test data selection methods, such as random selection because it simulates the typical work flow of a real user using the system. The trained classifier has to deal with the classification requests of a full day chronologically. Especially the MLkNN could perform in a misleading way when a test example is chronologically close to training data because as the labels were always applied for five sequent feature recordings. This behavior can be easily and successfully avoided by the method proposed.

6.4.2. Evaluation Metrics for Multi-Label Classification

In order to evaluate the success of a multi-label classifier, different metrics than in traditional single label problems are required. As defined by Tsoumakas and Katakis [84], D is the multi label data set, consisting of $|D|$ examples $(x_i, Y_i), i = 1..|D|, Y_i \subseteq L$ where L is the array of available labels. H is the multi-label classifier and Z_i the array of labels predicted by H for example x_i. For evaluation of the method several metrics have been calculated:

a) Hamming Loss: As defined by Schapire and Singer [77], the Hamming Loss evaluates how many times a label tuple is misclassified. Either the label was predicted wrongly or should have been predicted in the first place. A smaller value implies a better performance. Δ corresponds to the binary XOR operation and stands for the difference of the two arrays, calculated column-wise. This way a distance similar to the Manhattan distance as defined in [12] is computed.

$$\text{HammingLoss}(H, D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \Delta Z_i|}{|L|} \tag{6.1}$$

b) Accuracy, Precision, Recall: As proposed by Godbole and Sarawagi [35], these values were calculated. Similar to the Hamming Loss the Accuracy value indicates how exact the actual classes and the prediction matched. Detailed information as to whether the classifier is correctly predicting labels is given by the Precision value. This characteristic number gets lower the more labels were falsely predicted as positive regardless of how many labels are falsely negative. The Recall value is at the maximum of 1.0 when every label of the instance has been matched, regardless of how many other labels were predicted falsely positively. So this characteristic value determines how many of all active classes were classified correctly.

$$\text{Accuracy}(H, D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|} \tag{6.2}$$

$$\text{Precision}(H, D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \cap Z_i|}{|Z_i|} \tag{6.3}$$

$$\text{Recall}(H, D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \cap Z_i|}{|Y_i|} \tag{6.4}$$

c) One-Error: Evaluates how many times the highest ranked label (determined by $maxrank()$) is not part of the corresponding label array [95]. A One-Error of 0 means that one label of the instance is always matched with the predicted label with the highest confidence. This metric corresponds contrarily to the classical classification success rate of single label classifiers.

$$\text{OneError}(H, D) = \frac{1}{|D|} \sum_{i=1}^{|D|} maxrank(Z_i) \notin Y_i \tag{6.5}$$

Using these metrics, the results of different classification algorithms can be compared. Also, the performance among the participants can be analyzed.

6.4.3. Classification Results

The classification results are shown in table 6.3 for the MLkNN algorithm and in table 6.4 for the RAkEL method. For each user the average result over the full time of the study was calculated. Incomplete days with less than 40 entries (equals less than two hours of labeling time per day) were omitted. Below each table the mean, minimum, maximum, and standard deviation value over all users for each evaluation method is shown. Both methods perform very similarly. For some users the first algorithm works a little bit better, for others the second one. Overall, the quality of the results is the same and can be evaluated together.

The Hamming Loss range is from 0.1 to 0.2, which corresponds to a mean wrong result of one to two labels for each prediction. The accuracy partly corresponds contrarily to this value, which means that users with a low Hamming Loss show accuracy values up to 0.7. The large range of the precision and recall values hints at the fact that somehow the classification was right but in most cases not for the complete list of labels. The mostly higher precision than the recall value shows that the classifier more often classified a label falsely negatively than falsely positively. So one problem here is that the algorithms are underpredicting labels. This is especially true for the MLkNN classifier. The error of the predicted labels is less severe as displayed by the precision numbers. The average One-Error is about 38%, which means that in 62% of all predictions the label with the highest rating was guessed correctly. In general, the result fluctuates very much depending on the individual user. For some users the results are quite good but for others they are nearly unusable in order to predict the current labels from the recorded sensor data.

A closer look at the classification results gives a better insight as to which categories performed well and which can hardly be classified correctly. For this detailed view the data of the first user were selected because the overview showed promising results in general prediction success. The outcome can be seen in table 6.5. When the prediction for the label was correct, it is a "True Positive." When a "False Positive" happened, the classifier suggested a label but it did not match. The other way around, a "False Negative" means that the label should have been predicted. As a first result the RAkEL predicted 8,024 labels, while the MLkNN algorithm was a bit more conservative with only 7,524 labels during the study time. The in-depth view shows that the categories split up into three main categories ("Miscellaneous" was left out as it is unknown what exactly was labeled in this category and can therefore vary very much):

- Can be predicted very well with a success rate over 75% ("Work," "Spare Time")

Table 6.3.: Classification results for the MLkNN algorithm by user for the complete study time. Below the line minimum, maximum, mean, and standard deviation.

User	H-Loss	Acc.	Prec.	Recall	One-Err
#1	0.099	0.459	0.720	0.567	0.176
#2	0.076	0.692	0.746	0.710	0.270
#3	0.143	0.286	0.506	0.470	0.270
#4	0.179	0.167	0.410	0.254	0.533
#5	0.188	0.193	0.475	0.260	0.408
#6	0.182	0.188	0.498	0.272	0.394
#7	0.197	0.187	0.443	0.272	0.424
#8	0.187	0.175	0.493	0.266	0.357
#9	0.150	0.256	0.488	0.407	0.355
#10	0.173	0.202	0.465	0.295	0.439
#11	0.210	0.130	0.427	0.192	0.399
#12	0.147	0.297	0.537	0.436	0.279
#13	0.216	0.119	0.370	0.202	0.456
#14	0.153	0.223	0.566	0.291	0.344
#15	0.185	0.196	0.357	0.286	0.525
#16	0.119	0.446	0.559	0.467	0.455
Min.	**0.076**	**0.119**	**0.357**	**0.192**	**0.176**
Max.	**0.216**	**0.692**	**0.746**	**0.710**	**0.533**
Mean	**0.163**	**0.263**	**0.504**	**0.353**	**0.380**
SD	**0.038**	**0.145**	**0.104**	**0.139**	**0.094**

Table 6.4.: Classification results for the RAkEL algorithm by user for the complete study time. Below the line minimum, maximum, mean, and standard deviation.

User	H-Loss	Acc.	Prec.	Recall	One-Err
#1	0.101	0.419	0.653	0.607	0.167
#2	0.073	0.697	0.747	0.756	0.251
#3	0.157	0.234	0.507	0.410	0.291
#4	0.194	0.152	0.299	0.306	0.534
#5	0.206	0.141	0.353	0.284	0.413
#6	0.192	0.182	0.387	0.332	0.403
#7	0.193	0.192	0.411	0.332	0.390
#8	0.192	0.180	0.372	0.350	0.383
#9	0.164	0.226	0.403	0.408	0.393
#10	0.180	0.181	0.399	0.330	0.445
#11	0.228	0.098	0.298	0.239	0.410
#12	0.155	0.278	0.499	0.454	0.280
#13	0.244	0.070	0.276	0.198	0.460
#14	0.164	0.220	0.471	0.354	0.338
#15	0.177	0.240	0.406	0.365	0.450
#16	0.124	0.381	0.478	0.494	0.442
Min.	**0.073**	**0.070**	**0.276**	**0.198**	**0.167**
Max.	**0.244**	**0.697**	**0.747**	**0.756**	**0.534**
Mean	**0.172**	**0.243**	**0.435**	**0.389**	**0.378**
SD	**0.043**	**0.146**	**0.122**	**0.134**	**0.089**

Table 6.5.: Detailed results for user #1. Classification results listed for each category with corresponding results by MLkNN and RAkEL algorithm for the whole time period. Missing percent points are due to rounding errors.

MLkNN	Meals	Drinks	Hygi.	Work	Spare	House.	Social	Transp.	Misc.
True Pos	46 (9%)	2 (1%)	0 (0%)	1139 (76%)	1912 (75%)	0 (0%)	515 (27%)	173 (40%)	0 (0%)
False Pos	30 (6%)	2 (1%)	13 (9%)	200 (13%)	271 (11%)	0 (0%)	616 (32%)	54 (12%)	5 (7%)
False Neg	409 (84%)	273 (99%)	130 (91%)	156 (10%)	377 (15%)	130 (100%)	794 (41%)	207 (48%)	70 (93%)
Overall	485	277	143	1495	2560	130	1925	434	75

RAkEL	Meals	Drinks	Hygi.	Work	Spare	House.	Social	Transp.	Misc.
True Pos	106 (15%)	19 (4%)	19 (8%)	1174 (81%)	2005 (77%)	4 (2%)	600 (33%)	271 (58%)	10 (8%)
False Pos	230 (34%)	175 (39%)	109 (46%)	146 (10%)	317 (12%)	43 (25%)	521 (28%)	90 (19%)	60 (46%)
False Neg	349 (51%)	256 (57%)	111 (46%)	121 (8%)	284 (11%)	126 (73%)	709 (39%)	109 (23%)	60 (46%)
Overall	685	450	239	1441	2606	173	1830	470	130

- Very hard to predict successfully with a rate below 15% ("Meals," "Drinks," "Hygiene," "Household")
- Partial success ("Social Contacts," "Transportation")

Inspecting the details, the statement about the prediction behavior from above is shown. The algorithms clearly compute more false negatives than false positives as was already indicated by the precision and recall values. For activities of the first category, this ratio is about even but for the other two the classifier always predicted fewer labels than it should have. Referring to this behavior, the RAkEL algorithm performs a little bit better but along with that comes a higher falsely positive rate. On the one hand, the true positive rate of the RAkEL method is higher but, on the other hand, it also results in much more falsely positive label predictions. It cannot clearly be said which one is the better choice.

6.5. Refining the Classification Approach

Two improvements were implemented in order to enhance the results. On the one hand, the list of features was extended by more sophisticated features and, on the other hand, more classifiers were applied for testing. As proposed by Kwapisz et al. [50], additional accelerometer features were computed. This should help to classify some activity categories with specific movement characteristics. To emphasize the time, which is obviously important because it is directly correlated to activities, four more time features were added. Detailed information about the features can be found in the next chapter. Overall, the list of features was extended by 44 new features:

- Accelerometer: average absolute difference of each axis (x_{29}, x_{30}, x_{31})

Table 6.6.: Comparison of different classification methods. For each method the old and the new set of features was tested. The number in brackets after the MLkNN is the number of neighbors used in the run. The best value of each column is underlined.

Classifier		**H-Loss**	**Acc.**	**Prec.**	**Recall**	**One-Err**
MLkNN (5)	old	0.171	0.248	0.465	0.348	0.391
	new	0.166	0.258	0.491	0.348	0.386
MLkNN (10)	old	0.163	0.263	0.504	0.353	0.380
	new	0.160	0.270	0.528	0.344	0.375
MLkNN (20)	old	0.158	0.274	0.534	0.350	0.371
	new	0.156	0.282	0.545	0.354	0.366
MLkNN (35)	old	0.155	0.281	0.557	0.347	0.365
	new	0.154	0.289	0.557	0.358	0.361
MLkNN (50)	old	0.155	0.282	0.558	0.349	0.365
	new	0.154	<u>0.290</u>	0.561	0.359	0.361
MLkNN (75)	old	0.156	0.278	0.559	0.345	0.365
	new	0.154	0.288	0.564	0.358	0.358
MLkNN (100)	old	0.156	0.277	0.562	0.340	0.367
	new	<u>0.153</u>	0.288	<u>0.571</u>	0.356	0.358
RAkEL	old	0.172	0.243	0.435	0.389	0.378
	new	0.183	0.212	0.390	0.377	0.399
BPMLL	old	0.172	0.218	0.339	<u>0.487</u>	<u>0.340</u>
	new	0.183	0.198	0.313	0.472	0.366

- Accelerometer: average resultant acceleration (x_{32})
- Accelerometer: time between maximum peaks (x_{33}, x_{34}, x_{35})
- Accelerometer: time between minimum peaks (x_{36}, x_{37}, x_{38})
- Accelerometer: Binned Distribution (x_{39}-x_{48}, x_{49}-x_{58}, x_{59}-x_{68})
- Time: time of day (x_{69}, x_{69})
- Time: time of week (x_{70}, x_{71})

These new features were tested again against the MLkNN and RAkEL classifiers for review. Also, a new true multi-label classifier, the Back-Propagation Multi-Label Learning learner (BPMLL) [96], was added to the list. As the MLkNN was the most promising classifier here, the parameters were varied by testing different numbers of the nearest neighbors. The original MLkNN used 10 neighbors. The results can be seen in table 6.6. In each row the mean metric results for all participants are available. For direct comparison there are the results for the old features set in the top sub-row. In the bottom sub-row the results for the new feature set are depicted.

For the MLkNN the additional features generally improve the results. Not as well as expected but at least the classification metrics are no worse than the original results. The RAkEL and BPMLL classifiers do not benefit and show a higher error rate in all metrics when using the new feature set. The MLkNN classifier performs better the more neighbors are used. But starting from 35 neighbors, the results increase only marginally as seen in figure 6.12. With the use of the BPMLL classifier, Recall and One-Error values are better than with MLkNN. High values of these metrics can be achieved by overpredicting labels

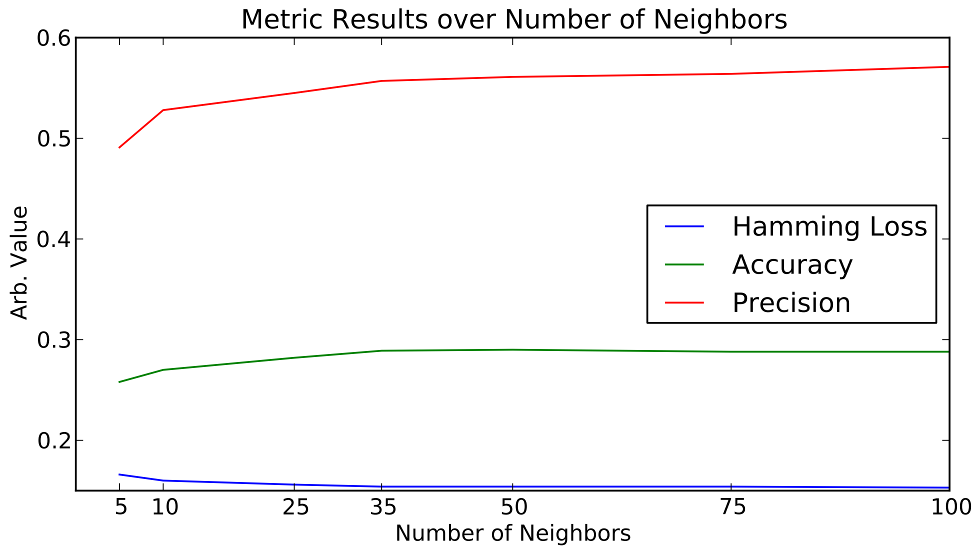


Figure 6.12.: Selected metrics results for MLkNN with varied number of neighbors.

and respecting the remaining metrics, this algorithm simply very often predicts false-positives, which leads to the good results in Recall and One-Error. Overall, the refining process did improve the results. The tests with different classifiers and parameters showed that the best choice is a MLkNN with 35 neighbors.

6.6. Discussion and Conclusion

During the course of the study a huge amount of data was collected for and by each participant. To get an insight into this complex and manifold data, datamining techniques were applied to restructure and condense the information. In a first step, the raw sensor results were used to compute features which are able to represent each time slot in which data were recorded. These features were visualized to get a first look at the daily activities. For each feature set the corresponding label is available holding the connected information of the activity category. Instead of reviewing the complete data set, one participant was picked out prototypically. A simple plot of all feature readings over the complete study time has already revealed first characteristics. The daily structure is visible and the location information gives clear hints for different location spots. But also the first problems can be identified: The GPS sensor did not always return the location and the network provider seems to rely on the last-known information when no network connectivity is available. In a next step, a principal component analysis limits 90% of the variance to the first five resulting components leading to the assumption that many of the features correlated with each other.

For an in-depth view, one day was inspected in great detail. Correlations between some main activities and sensor readings can be found and suggest success for later classification approaches. A detailed review was conducted in addition because especially the location seems to be very important. Time and activity categories can be correlated to specific locations and will be a strong indicator for the classifier. Also the routes can be identified and this way connected to activities. A second participant confirms this finding with similar behavior. For the remaining features no evident correlations can be found for most of the categories. Only the "Transportation“ activity can be identified by high accelerometer and sound amplitude sensor readings. As the user was allowed to enter more than one activity for a time window, the system has to offer the functionality of entering these combinations. Also the transitions between activity categories reveal some information about the typical behavior of the users.

When looking at the previous findings, the classifier of choice has to be able to classify combined activities. This is the reason a multi-label classifier is needed. Mulan, a Weka extension, provides easy implementation and testing of multi-label classifiers and was used. Two suitable methods were presented, MLkNN is a true multi-label classifier, and the RAkEL method handles the multi-label data by computing permutations which then can be used with a single-label algorithm. For realistic training and test data, the leave-one-out method was applied, which ensures no side-effects in combination with the MLkNN classifier. The results show that relatively often there are errors in the prediction. A detailed look reveals that this strongly depends on the categories. For some the classifiers work very well and others are nearly impossible to predict. Overall, the MLkNN performed better than the RAkEL method. A refining process by adding more features and testing more classifiers and parameters enhanced the results slightly. The best working classifier is the MLkNN method when using 35 neighbors as the parameter.

The set of categories was defined before the study had started without the knowledge of how well they could be classified. The selection was based on a pre-study and reflected the needs of typical users. Therefore, it is not surprising that some categories are very hard to detect from sensor data only. There are especially activities which cannot be assigned to any sensors integrated into the phone. It is impossible to detect "Drinks“ or "Hygiene“ when there is not a special situation the user is in every time the activity is performed. For other categories, such as "Work,“ the selected methods worked very well, assumingly based on the location data. But even for others like "Transportation“ the classifier could achieve positive results. Overall, the result very much depended on the category. The choice of activities has to be adapted in order to gain better results. This also would reflect the highly individual behavior of the users who may prefer other sets of activities where their personal interests are respected.

One lesson learned for a future study is that the devices need mobile internet access. Not for transferring results or any other experiment data but for obtaining correct location data. On the one hand, the network location provider needs a connection to an online service and, on the other hand, the time to the first fix of the GPS sensor can be dramatically decreased with online access to assisting data (AGPS). As the location is an important indicator for the activities, here the results can be improved by increasing the quality of the sensor data. In general, the capturing process itself works very well and the devices last a full day without any problems when collecting data every 3 minutes for a time span of 20 seconds. The computed sensors did not decrease the classification rates after the refining process, which means that they can be kept for the next cycle.

The first study showed that smartphones are able to collect the data required for activity classification, which is needed to support the user in the original task of keeping a digital diary. The tools and methods to work with daily activity data are proven to work and can be applied in future scenarios. Respecting the fact that the results may not be perfect, it can be very well used to support the user with suggestions. The outcome of the classifier is a valuable hint at which activity the user is doing at the moment in question and can provide suggestions for the current situation. This suggestion is the basis for a faster and easier way of recording activities because they do not have to be selected manually. The ongoing work must concentrate now on how the user can be supported in entering activities and how suggestions can be provided in order to minimize the time and effort to keep a digital diary of activities. Keeping this in mind, the next generation can be developed. This system will be presented in the following chapter.

7. Development of the AMARAS System

The *Adaptive Multi-Modal Activity Recognition Assistant for Smartphones* (AMARAS) is an integrated demonstrator application which supports users in daily activity tracking. The previous study has shown that a smartphone is able to collect sensor data that can be used to guess or at least provide a hint for the current activities of the users. The data were collected directly on the phone and processed and evaluated in a separate offline step. The original goal to collect meaningful data without any extra hardware was achieved without limitations. The current chapter is about the next development cycle of the application that shows that it is possible to integrate a complete system which assists users in on-device activity tracking. A new application was designed and implemented (cf. research question 3) and will be tested in another study. Overall, AMARAS will be used to show that the initial idea of assisted, diary-like activity tracking with smartphones can be used to support people in everyday life situations.

7.1. Re-defined Concepts and Goals

The goal of this thesis is to show that smartphones are able to support users in diary-like activity tracking. As AMARAS has to fulfill this expectation, the objectives for the demonstrator planned have to be clearly defined. These have to meet all the previous specifications allowing the application to provide proper experimental data to support the results that are expected.

The first of these objectives is to keep the smartphone as the only hardware used for this demonstrator. No external hardware should be required or have to be attached. The application only works with data that can be collected from internal resources. The previous study showed that a perfect classification of activities is not possible, which in turn means that this is not the goal of this application either. Instead it aims at supporting the activity tracking process as well as possible. This is achieved by providing predictions for the current activities. Only in case the situation can be clearly identified does it store the results automatically. If not, the users have to be asked to provide more information. It is very important to collect and save correct data only. In case the users are interrupted, it is important to make the activity selection process as fast and as efficient as possible. By preselecting and pre-ordering the available activities this goal is will achieved. Overall, the application planned here tries to figure out the current situation first, and only if it is not confident enough about the situation, does it ask the user for additional input. It then shows the activities sorted by relevance and with the best estimates already selected which then just have to be confirmed - ideally with a single click.

Another result of the first study is that activities are highly individual and depend on the operating user. It is not possible to cover all available activities or even to create a set fit for a specific user group. This leads to the fact that the users need to be able to create a customized list of activities on their own. The planned application will no longer provide a static set of categories and activities. Each user can decide which granularity, number and type the preferred activities will have. On the one hand, users can decide to track only very coarse activities, e.g. if they just want to know the ratio of their work time compared to their free time, while, on the other hand, they can choose to record their day in a much more detailed way. The application has to provide functionality for this individual behavior and has to be able to deal with any type of activity. Therefore, a general approach for predicting and storing activities of the daily life is needed. The need for the ability to select multiple activities at the same time still exists and has to be respected in the planned application as well.

Starting where the last implementation ends, the smartphone is able to collect and store sensor data. Storage and battery life are sufficient and the platform enables data collection without hurdles based on the direct access to raw sensor data. In addition to the data collection, this time the application also has to process and evaluate the collected information. Theoretically this task could be outsourced to an external service but an on-device solution provides a maximum amount of data protection and privacy. The sensitive data always stays on the device and cannot be viewed, copied, or stolen without physical access to the device. It theoretically works completely autonomously without any network or internet access. On the downside, the new approach needs much more processing time, which has to be considered concerning battery life. It should still at least last a normal full day of a user, which is, according to the previous study, about 17 hours long.

Summarizing the developed concepts and goals, AMARAS is completely self-contained. An on-device classifier is trained with collected data of the user and provides estimates about current activities. Depending on the result, the system decides which action to take and whether the user has to be asked for more information on the current situation, or if it can detect and store the activities automatically. This process starts with collecting sensor data. These data are then used to extract meaningful features to be evaluated by the classifier. The classifier is trained with the resulting set of features. As training a classifier is a time consuming process, the time of day at which to perform this task must be considered carefully. Once the classifier has been trained, it can be used to test feature data in order to extrapolate activities.

When dealing with a machine learning system it is crucial to ensure that high quality training data are available. AMARAS is able to fill in activities automatically without any user interaction. This may lead to incorrect data if there is a misclassification. It then becomes even worse in the next repetition of training with these wrong data. In order to avoid the corruption of the classifier, a separate review process is proposed. In this step the user has to confirm all the activities that were entered automatically or manually. This also allows for the ability of correcting accidentally wrongly entered activities or add forgotten ones later on. The review process ensures that the classifier gets trained with correct data only and is an important prerequisite for good future predictions.

In summary, AMARAS can be divided into four major parts. The first one is to extract features from recorded sensor data on a regularly basis. Secondly, the user has to be asked about his current activities from time to time in order to collect labels for the recorded data. Additionally, there will be a way to enter activities manually if they change in the half way through the asking intervals. In the third step, high quality label data are ensured by reviewing all entered activities once. This can happen at user request but at least before the next step: Here the classifier is trained with existing feature data combined with corresponding reviewed activity data. The classifier only deals with reviewed data. Therefore, the review step should happen before the training begins. As already said training can take a long time and this costs battery life on smartphones. This is no problem when connected to a wall charger, and this subsequently leads to the conclusion that training will be done at night and only if the phone is connected to a power source. As a side effect this is a nice equivalent to human dreaming, where experiences of the day are processed in the brain. Once the classifier is trained, predictions can be made upon request. By time the classifier learns about the activities of the user and classification, results will hopefully improve.

AMARAS is self-contained and does not rely on any kind of external hardware. It is able to support the user by providing predictions on the current activities. Users can build up the list of activities individually, and a general-purpose classifier is able to deal with any kind of labels. Of course, the quality of the results depends on the activities that are chosen to be tracked; not all activities will be possible to detect with the given set of built-in sensors in the smartphone. To test if this works as expected and what kinds of problems may occur, the proposed system needs to be designed, implemented, and evaluated. The biggest challenge here is to develop a system which runs properly with the limited resources of a smartphone.

7.2. Application Implementation Details

Like the application that was used in the previous study this one will also be implemented to run on Google Android smartphones. Although the phones are now two years old and not up-to-date anymore, they will still fulfill the needs of the planned system. Meanwhile, a firmware update for the smartphones has been released which fixes various problems and bugs. Most noteworthy is the fact that the phone is now able to collect accelerometer and other sensor data while the screen is off. Before, the screen had to be turned on by a system call in order to get reliable data. This enables the application to run completely silently, and the users will not notice anything while the phone records data. The new firmware is identified as “Android Gingerbread 2.3.3” and was officially released by HTC.

A supervised bachelor thesis by Tobias Rodehutskors has shown that it is possible to run a classifier directly on the phone [74]. In his work Rodehutskors used the Weka framework to distinguish different situations only by analyzing sound that can be recorded using the built-in microphone of a smartphone. This also adds a new and valuable sensor source to the planned system. As AMARAS will be much more complex, the old application can only be reused partially. A completely new design was developed and needs to be implemented starting from scratch. The new application needs to be modular and extensible to fit all the required and future needs. The previous results have shown that an MLkNN classifier works best and, therefore, this will also be the choice for the upcoming application. As an additional requirement, the application has to provide functions to easily change the classifier later. This ensures that the classifier can be changed later if there are any unforeseen problems.

Note by the author: While the initial work, planning and architecture of the in the following described application was done by the author, the software implementation was

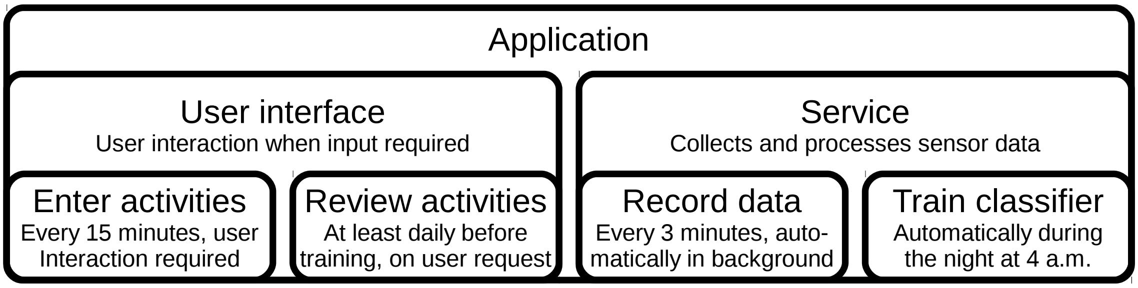


Figure 7.1.: Application overview. The application is separated into a user interface and a service part. Every time user interaction is required, the UI is called and delivers the input needed. In the background the service runs autonomously and automatically collects and processes sensor data. This process is completely invisible to the user.

done by the student assistant Tobias Rodehutskors. With the knowledge he gained during his bachelor thesis he was the ideal candidate for this project. At this point I would like to thank him for his great work! It was a pleasure working with him and I could not imagine anyone who could have been a better fit for this task. Without your outstanding self-reliance this work would not have been possible in this short amount of time. Thank you very much!

7.2.1. Application Overview

Logically the application is separated into two main parts:

Application Service: For all tasks not visible to the users an independent service runs in the background (right half of figure 7.1). It manages the data recording and processing, and it stores everything into a database. Also, the service is in charge of training the classifier. The service is implemented by using an *Android Service*[1] and runs continuously.

User Interface (UI): Every time the application needs information which can only be provided by the user, the user interface (left side of figure 7.1) is called and brought up to the front. The two main tasks are to retrieve activity information manually and start the review process in order to verify activity data. The user interface is invoked by the service automatically by timers or can be started manually from the launcher if needed.

The service and the user interface are connected and communicate with each other. If the service crashes for any reason, it will be restarted automatically by the operating system. This ensures continuous operating and also decouples large parts of the application from the main application user interface. This increases stability and reliability for the long running application.

In normal mode the application runs, records data, and collects daily activity labels. In order to provide comfort and more control for the user, there are two additional modes as

[1] `http://developer.android.com/reference/android/app/Service.html`

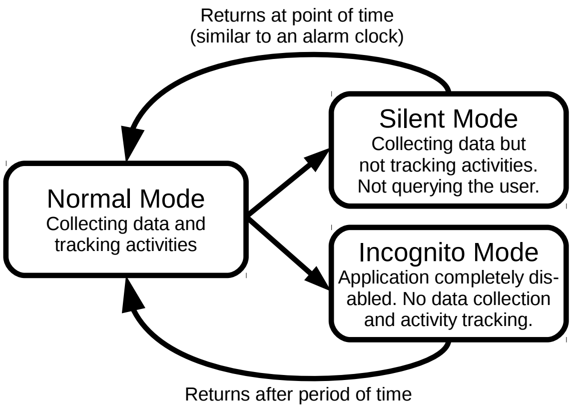


Figure 7.2.: Application modes. In silent mode the data recording is still running but there are no interruptions for the user. In incognito mode the application is completely disabled. No data are recorded at all. It is used to return to normal mode after a specific amount of time.

also seen in figure 7.2: The first mode is the “silent mode.” Here, only the user interaction is disabled. Everything else still runs like normal. This mode is for meetings, activities where the user does not like interruptions, and can also be used to silence the phone when the user is asleep. The second one is the “incognito mode.” Here, the system stops any kind of data recording and user interaction. The service is on hold and is only kept alive to deactivate the incognito mode after a certain amount of time. It cannot be activated without setting an ending time. This ensures that the user cannot forget to disable it. This mode can be used if the user does not want that the application records anything. In contrast to the silent mode the user is not asked to enter a point of time similar to an alarm clock but a period of time when it ends. In general the application itself cannot be stopped by the user. These two modes offer control for the user while ensuring that accidental user input will not stop the app for a longer time during usage.

7.2.2. Background Service

The background service is all about timing and starting the related tasks. Three processes have to be timed, started, and stopped in the background. Any kinds of data that is recorded, calculated, or put in by the users are stored in a database.

Recording Data: Like with the previous application, data are recorded every three minutes. This interval is not fixed to a certain point of time but starts automatically after booting the smartphone. After the start has been completed, the interval of three minutes is initialized. Again battery life is the main reason not to decrease this time. Every time this alarm is triggered, the recording process starts. The complete cycle is shown in figure 7.3. If the incognito mode is activated, this process ends immediately. If not, four phases are scheduled. The preparation phase begins with an early preparation process that activates sensors that need a certain amount of time before working correctly. It starts 30 seconds before recording begins. Mainly it is the GPS sensor that tries to get

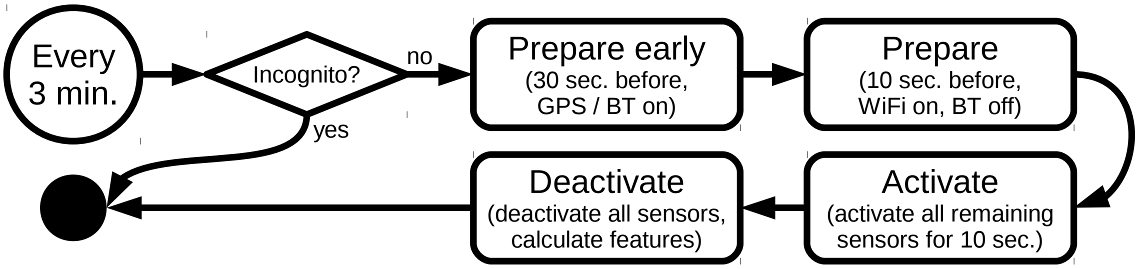


Figure 7.3.: Data are collected every three minutes (from the time of application start) if not in incognito mode. In a first, step GPS is activated because it takes some time until a position has been estimated. Bluetooth cannot be used at the same time as WiFi, which means that it scans first. In the next phase, WiFi turns on and Bluetooth off. Another 10 seconds later the recording process starts and collects data for 10 seconds. The process ends with deactivating all sensors and extracting the features for the raw data .

valid location information during this time. As Bluetooth and WiFi interfere when used at the same time, the early preparation state is also used to scan for Bluetooth devices nearby. In a second preparation state, 10 seconds before recording, the Bluetooth module is stopped and the WiFi sensor is activated (if not already active). The activate-state then starts all sensors and records data for 10 seconds. Afterwards every module is stopped and disabled again waiting for the next time the process is invoked by the service.

Entering / Predicting Activities: In order to get label information that can be assigned to recorded sensor data, the user has to enter the activities for a specific time frame. These data are used to train the classifier, which will support this task by activity label suggestions once trained. Even though the last study has shown that subjects would prefer a longer interval, the same time span was chosen again to get accurate data. Ideally, the user should enter activities every time a new one is started. Hopefully the use of the classifier will increase the time between interruptions by providing completely automatic guesses or at least shorten the time by smart suggestions.

As seen in figure 7.4, the user will not be asked for activities if the application is in incognito or silent mode. In both cases the process will terminate without any action. In normal mode the first step is to use the trained classifier (if available) to guess the activities since the last time the user entered any. For each time sensor data was recorded, the classifier predicts the corresponding activity confidences. The mean value of all predictions for each individual activity label is used to decide if a fully automatic selection of activities can be made. If this is the case the results are stored and the user interface has not to be called. For completely unsupervised activity selection the confidence for each activity has to be above or below a certain threshold. In the case that user interaction is required the user interface is called. The user now can select the activities for the current time frame. Again the mean results of the classifier are used to sort the list of activities and another threshold automatically preselects the most likely activities based on the confidence. The confirmation stores the data and ends this process.

In order to illustrate the triggering of the 3-minute and the 15-minute process, the cycle has been visualized in figure 7.5. While the 15-minute process is fixed and starts with the

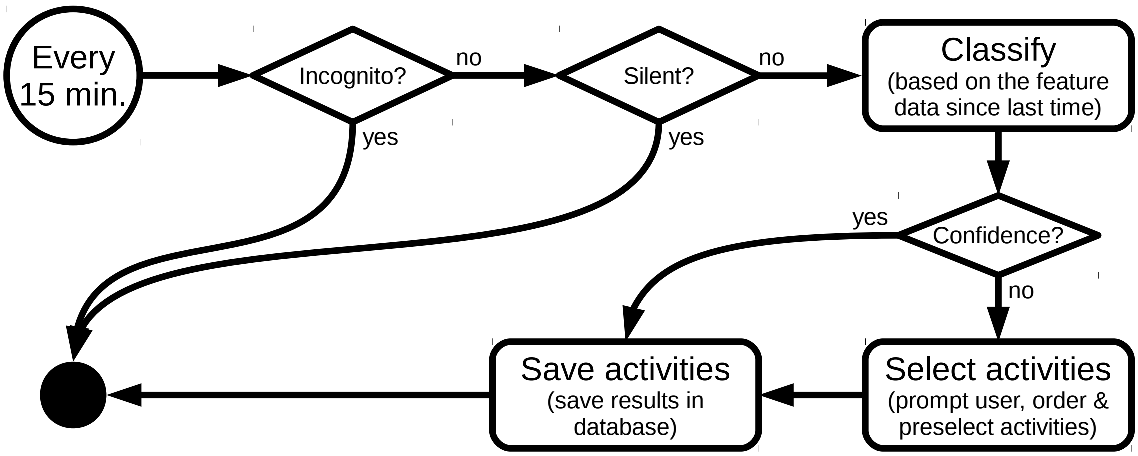


Figure 7.4.: Every 15 minutes (fixed quarterly interval) an alarm triggers the system to ask the user for the activities which have taken since the last time of asking. If it is not in incognito or silent mode, it uses the feature data collected since the last time of asking and uses the classifier to predict the current activities. If the activities can be clearly predicted, the result is saved directly into the database. If not, the user is asked to take action and enter activities. Based on the results the activity list is pre-ordered and preselected based on the confidence based for each label.

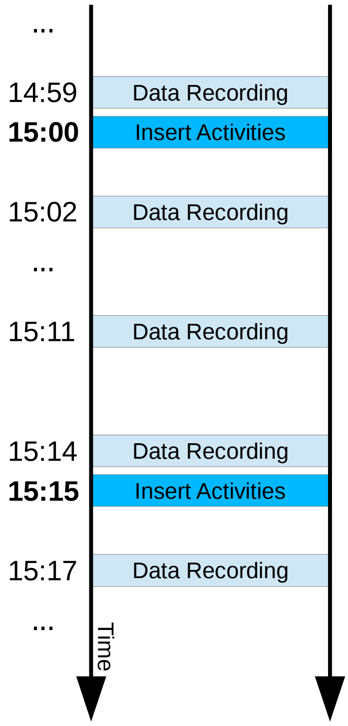


Figure 7.5.: Alarm timing for data recording and inserting activities. While the activity insertion process is fixed to a quarterly interval starting with the full hour, the recording alarm is triggered every three minutes starting with the launch of the application. Therefore, the actual point of time varies.

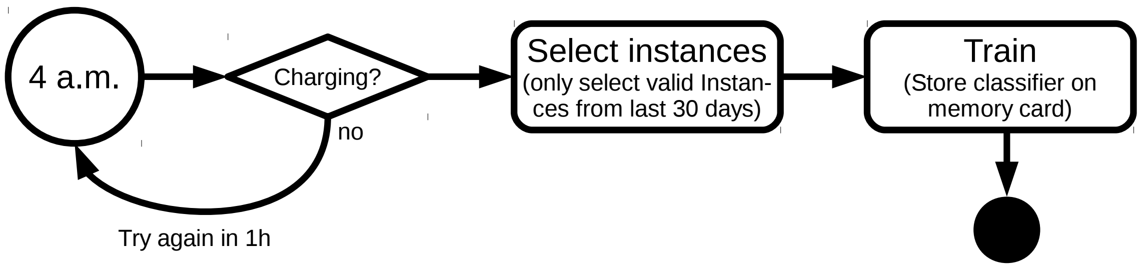


Figure 7.6.: The application updates the classifier every night. The training starts only if the phone is connected to a charger. After selecting and filtering instances from the last 30 days, the classifier is trained and stored on the memory card. If not charging, the next attempt is made one hour later.

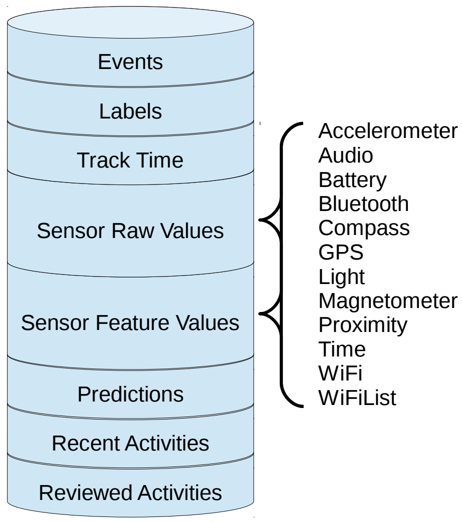


Figure 7.7.: Structure of the database. Sensor raw and feature values are available for all sensors with their corresponding set of values. The data are stored using a SQLite database on the external memory card.

full hour, the 3-minute process starting time varies depending on the application start. In this case it starts from 14:59 and repeats itself every three minutes. This dynamic approach prevents the recording interval from having a fixed temporal distance to the activity labeling and thus offers more variation. Additionally, the recording process is stopped and started by the training process every night. As this varies in length, the recording process has a different starting time every day.

Training the classifier: Every night the classifier is trained with the saved feature and activity label data. As the training process is very power consuming it is only started if the phone is connected to a charger as seen in figure 7.6. There is a cut-off of 30 days to limit resources. This also ensures that it adapts to recent changes in the daily activities. The trained classifier is serialized and stored on the memory card and can be loaded at startup without the need of initializing and training it again.

Database structure: Every data recorded by the application is stored in one single database located on the memory card of the smartphone. The SQLite[2] implementation of Android[3] is used to create and access it. The application can be updated without deleting the database and, therefore, data can be transferred between devices without data loss. Using a database offers fast queries and protects the data against corruption by automatically using a journal and other advanced internal functions.

As seen in figure 7.7, the largest part of the database is filled with raw sensor and feature data. The raw values are not needed anymore once the features have been extracted, yet these data could be useful for later evaluation and are kept. For each sensor there is a corresponding table used to store the individual values grouped in recording tracks. To evaluate the usage of the application, several kinds of actions, namely what happens in the service or when does the user interact with the applications, are stored in the "events" table. In order to decouple the label names, a list with IDs is maintained. The predictions of the classifier during the usage of the application are also stored for later analysis. Two tables for user activities help implement the review process. Every time the user enters any activities, they are first stored in the "Recent Activities" table. These data are used for the review process and have to be confirmed by the user. The result is stored in "Reviewed Activities" when finished.

7.2.3. User Interface

If the application was launched manually from the Android launcher, the main screen is displayed. This is the starting point for all user interaction. An overview of the complete user interface is visualized in figure 7.8. Note that the complete application is in German as the targeted user group for the planned study was Germans only. At the top, there are two buttons for enabling the incognito and the silent mode. While enabled the selected mode can be disabled early by a second click.

The largest button at the top center leads to the activity selection screen (figure 7.8 center row, center image). It offers the pre-ordered list of activities where the most likely ones are preselected based on the results of the classifier since the last insertion. Behind each activity name there is the corresponding confidence value displayed in brackets. If there are more than six activities stored, only the first five are displayed and the rest can be accessed through an extra button beneath the list. New activities can be added directly from this screen (figure 7.8 center row, right image). The time the activities will be assigned to is pre-configured to the time since last insertion, but this can also be customized. The selection process must be confirmed or canceled in order to get back to the main screen.

For the review process the user has two options: The saved activities can be confirmed directly on the phone or externally using a computer. The second option is presented later because it is more complex. When doing the review on the phone, the user will get a list of all entered activities since the last review (figure 7.8 bottom row, left). This list can be altered by deleting and adding activities in each time frame. A separate screen is available for longer and combined activities. After the review the data have to be confirmed to get back to the main screen. Now these activity data are used in the next training process.

[2] `http://sqlite.org`

[3] `http://developer.android.com/reference/android/database/sqlite/package-summary.html`

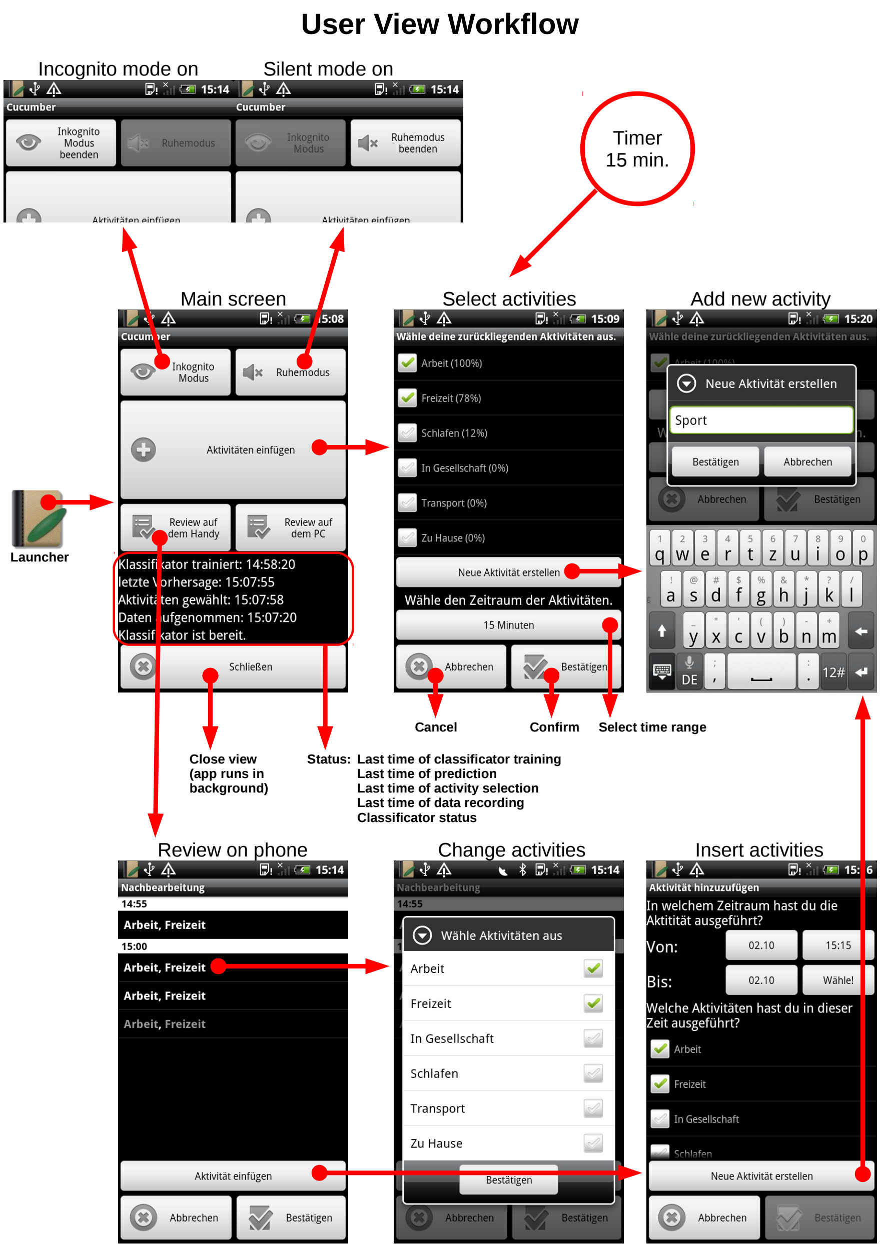


Figure 7.8.: After launching the application, the main screen is displayed. The data collection or the asking for recent activities is paused by tapping the button for the incognito or the silent mode. To insert activities, the large center button can be used. The same screen is also automatically called by the service when the 15-minute timer triggers. Activities are rated, preselected, and sorted by confidence and new items can be added directly. For the review process all recorded activities since the last review are shown and can be manipulated. The user interface can be dismissed but the service will not stop and constantly runs in background.

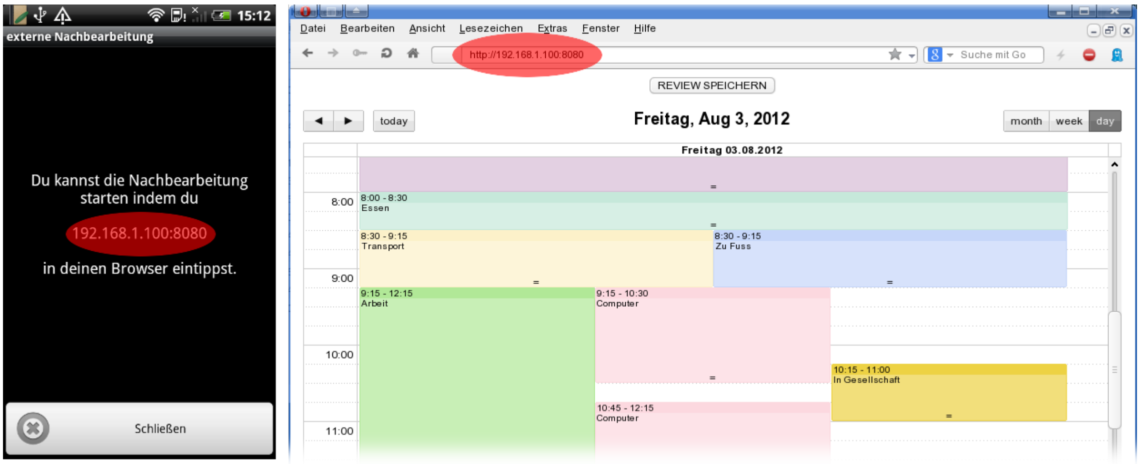


Figure 7.9.: The external review allows to confirm saved activities very comfortably on a computer within a browser. The target URL is displayed on the phone (highlighted in red) and has to be opened on the computer. In a calendar-like view the past activities can be altered and reviewed.

At the bottom of the main screen status information can be found for determining whether the application is running properly. To close the user interface there is a button at the bottom. Note: The service is still running in background, and the application cannot be shut down completely by the user. For debugging there is a hidden password protected entry in the menu, which shuts down the whole application.

The review process can be very time-consuming when done on the phone, and for this reason the application provides a so-called external review. This enables the user to review the activities on his computer where a larger screen makes it much more comfortable. Technically the phone starts a small webserver and provides the data so they can be accessed from a browser as seen in figure 7.9 on the right. In order to establish a direct connection between the computer and the smartphone, both have to be in the same WiFi network (exactly: same subnet). This method also provides a maximum of privacy as no third-party server is required and could also be encrypted and password protected.

The user starts the external review by clicking the button on the smartphone. The network address the user has to enter in the browser on the desktop computer appears on the display and the activities since the last review will appear in a calendar-like view. The entries can be rearranged via drag&drop, deleted, and new activities can be created. The process can be finished in the browser and after the data are transferred back to the smartphone the review process is completed. Especially for a longer time period, e.g. in the evening after one full day, this method provides a very effective way of reviewing all activities.

7.3. Feature Extraction from Raw Sensor Data

The recorded data have to be processed first in order to extract meaningful features and to then be able to train the classifier and predict activities from smartphone sensors. With the trained classifier unknown feature sets can be classified in order to predict activities.

Each sensor built into the smartphone is accessed by a *sensor processor*. It encapsulates the access to the sensor data from the Android API and is managed by the *sensor manager*. This modular design enables an abstract handling of the sensors and provides comparability for future devices and tasks. Specific sensors can be disabled individually and *multi processors* can be designed easily combining more than one processor. This will become important for some features later. Each process of recording sensor data is called a track. After retrieving the raw values for a track, each sensor processor stores its extracted feature values in the database.

In comparison to the first application prototype the list of features has been extended and also new sensors have been added as data sources. An overview of all features is available in figure 7.10 grouped into sensor classes which each provides several raw values. These values are used to extract meaningful features, which then are fed into the classifier. The application utilizes 10 sensors and extracts a set of 102 features. Newly added sensors and features are marked with a small plus sign in figure 7.10. For some features a combination of sensors is needed and not all sensors are finally used, but the raw values are saved as they could contain important information for later analysis. Here are the features in detail, subdivided in groups by their providing sensor with their total number of features in brackets:

Accelerometer (46): Aside from very simple features like the mean value over time and the standard deviation additional features as mentioned in the work of Kwapisz et al. [50] were extracted (ED = example duration):

- Average Absolute Difference: Average absolute difference between the value of each of the readings within the ED and the mean value over those values (for each axis)
- Average Resultant Acceleration: Average of the square roots of the sum of the values of each axis squared $\sqrt{x_i^2 + y_i^2 + z_i^2}$ over the ED
- Time Between Peaks: Time in milliseconds between peaks in the sinusoidal waves associated with most activities (for each axis)
- Binned Distribution: We determine the range of values for each axis (maximum – minimum), divide this range into 10 equal sized bins, and then record which fraction of the values fell within each of the bins.

Microphone (15): The main features derived from the microphone input are MFCCs[4] as introduced by Logan [59]. These coefficients are a very compact representation of the power spectrum of the recorded sound and therefore ideal for the targeted application. A previous work done by Tobias Rodehutskors [74] has proven that they are eligible for sound analysis on a smartphone. Sound scenes can be distinguished and classified with the help of MFCCs as features. Additionally, the root mean square and the zero-crossings for the signal are calculated and used for additional information about the signal.

As it is illegal in Germany to record conversations without the knowledge and agreement of all attendees (§ 201 StGB), it is very important not to save the raw sound signal. Additionally, no method may be used which saves the recorded data in a form that could

[4] Mel Frequency Cepstral Coefficients

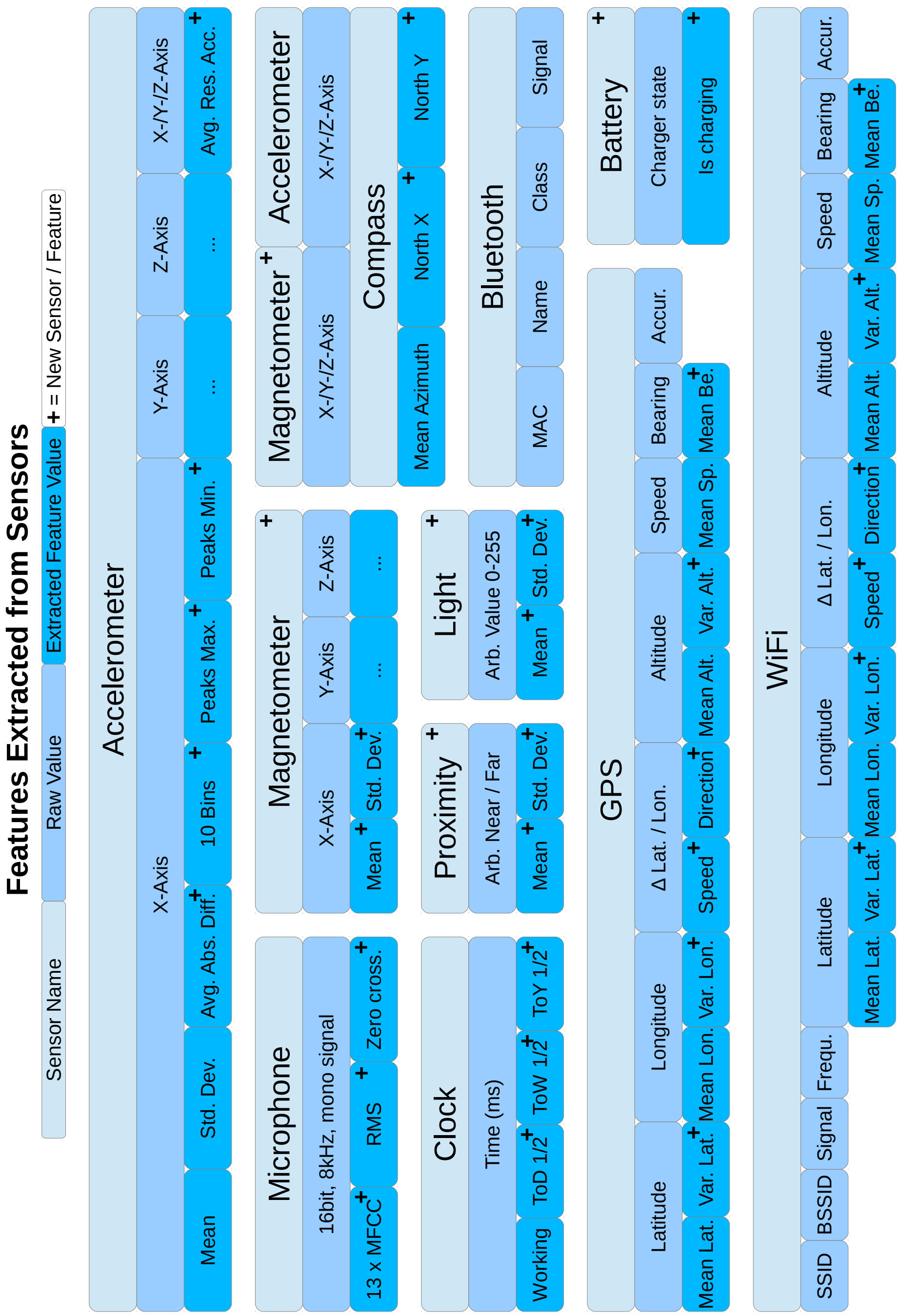


Figure 7.10.: Set of all extracted features of AMARAS from available smartphone sensors. Each sensor (at top, colored in light blue) provides raw data as shown in the second row of each set. Based on these data features are extracted. Overall, the result is a set of 102 feature values for each track.

be decoded in any way. This does not apply to the features selected but, unfortunately, this requirement limits the use of the raw data for later evaluation. Tests have shown that it would be possible to clearly recognize sounds, voices, and even far away conversation when recorded by the smartphone inside a pocket. This, of course, would be a huge privacy breach for the user and even for uninvolved bystanders, which is not acceptable. Finally, the sound is processed directly without even temporarily saving it.

Magnetometer (9): The magnetometer built-in in the smartphone measures the magnetic force in three orthogonal axes. For these values the mean value over time and the standard deviation are used as features. With the additional information of the current orientation of the phone by adding the accelerometer readings, a compass can be simulated. Aside from the mean azimuth, the sinus and the co-sinus values are calculated. These features provide information about the direction the user is facing during the record time.

Clock (7): The main problem with time features is that the distance between two time stamps has to be preserved. Especially when days, weeks, and months change this causes trouble. To overcome this fact, a time feature implementation as proposed by Leichsenring [57] was applied. In addition to the six resulting time features for daily activities, it is important to know if the current day is a workday, which is represented as the seventh feature.

Proximity / Light (2 / 2): The proximity sensor located near the earpiece might give a clue as to where the phone is located, e.g. inside or outside the pocket of the user. The same applies to the light sensor.

Bluetooth (0): Even though the Bluetooth devices in the vicinity are not used as a feature for the classifier, it could contain important information for later evaluation. It is not included because, on the one hand, it is not trivial to extract useful features and, on the other hand, the number of devices with activated, visible Bluetooth is very low.

Battery (1): Whether the phone is connected to a wall charger or not is a strong indicator for the sleeping activity and also hints at the fact that the phone is not located inside the pocket, which makes accelerometer feature less useful for activity recognition.

GPS (10): The global positioning system provides accurate location information outside of buildings. Aside from the mean and variance values for latitude, longitude, altitude, and speed, the mean direction over the recording time is calculated and used as a feature. Additionally, the speed and direction compared to the previously measurement point hints at where the user is heading.

WiFi / Network (10): Similar to the GPS sensor, the IDs of WiFi and GSM networks are used to determine the current location. In contrast here the data are not as exact as with GPS but it works inside buildings, too. Therefore, this is a coarse but reliable location source. The extracted features are the same as for the GPS sensor. For later evaluation the raw data also include information about the WiFi networks available at the time of recording.

All features of the previously built application were adopted or replaced by more sophisticated ones. The new list of features is the result of further research, findings, and developments during the analysis of the last results. The selection of features and the fact that they are computed directly on the device also mattered. Taking this into account, only features with a low complexity can be used as computing time results in a reduced battery life. For later analysis with more complex features the raw data is still available.

7.4. On-Device Classification of Daily Activities

Tests with data from the previous study have shown that the MLkNN classifier works best when it comes to detect daily activities from the originally provided feature set. This is the why reason the same classifier will be used in AMARAS, too. Although the number of features has increased, the overall data are comparable and, therefore, similar results can be expected. Additionally, the algorithm has been proven to run well on the selected smartphones as it was used in the bachelor thesis of Rodehutskors [74]. Some details as to how the algorithm works as originally proposed in the paper by Min-Ling Zhang and Zhi-Hua Zhou [95] are presented here for further information:

The MLkNN algorithm is derived from the traditional *K-Nearest Neighbor* algorithm. By utilizing the maximum a-posteriori (MAP) principle for the neighbors of the unseen instance, the likeliness for each label is calculated. In order to apply the MAP principle, the prior probabilities and the posterior probabilities are needed. These can be estimated from the training set beforehand based on frequency counting. This is the actual training phase of the classifier. For classifying the label set of a given instance, the estimated probabilities are used with the Bayesian rule. Additionally, a real-valued vector is calculated to rank the labels.

Previous tests have shown that the quality of the classification increases with the number of neighbors used with this classifier. As AMARAS should be able to learn new activities quickly, the number of neighbors cannot be set too high. A value of 25 neighbors was selected, which corresponds to a minimum of the same number of examples per activity. This leads at least to a time span of 75 minutes (recording every three minutes $\times$ 25 examples), which is needed for each label before the results are comparable. Therefore, in the typical 15-minute-interval of labeling the user needs to label the activity at least five times. In real world usage this number might be higher as not every recording example will represent the activity perfectly necessitating more examples to be on par with other labels. Using these settings the number of neighbors is high enough to expect good results, but also enables a fast learning of new activities or activities that do not last long but also are important in the daily routines. For example this could be “Transportation” while on the way to and from work.

By choosing the MLkNN classifier, an algorithm has been selected which has been proven not only to perform well on the smartphone but also to bring good results. It is available with the Mulan [85] software library, which is an extension to the Weka [38] software package. Again the software is built to be modular and the classifier can be exchanged easily for testing and special cases where others might perform better. Preliminary test were able to show that the system should be very well prepared for the upcoming task, especially considering the choices that were made in this regard.

7.5. Technical Details

The application only uses official documented Android API calls and, for this reason, should be able to run on any Android device. It is not required to root or modify a device in any way. Although the application will generally not interfere with other applications, it may limit the use on productive devices as battery life is drastically reduced compared to normal operation because of the repeated recording of data and feature extraction. The application is designed to run continuously and listens to the memory card mount event of the system in order to start itself after a restart of the phone. The Android service is set to restart itself automatically once stopped by a crash or by the system. In the worst case, the user can simply reboot the phone to clean everything and continue using the application. Any task that needs a longer amount of time is separated into a individual thread to keep the main application operational. For timing, repeating *Android Alarms*[5] are used.

The last study and other tests have shown that the current location can be estimated faster and more precisely if a network connection is available. While WiFi or network localization only work when the device is connected to the internet, GPS localization benefits as well because it is able to speed up the localization process by using the assisted GPS feature. This method also requires a connection to a remote server to retrieve the supporting data. As the current position is an important cue for the current activities, the smartphones should be equipped with a mobile internet connection. This also enables the phone to synchronize its clock using a timeserver. The correct time is crucial for data analysis because every entry in the database is saved not only together with an individual id, but additionally also the current timestamp. This is needed to connect events, raw data, features, and classification results.

The software library *Crittercism*[6] was integrated for monitoring and automatic crash reports. This helper software library collects application crashes on the devices and as soon it is able to connect to the server these reports are uploaded. The developer is notified via e-mail and can use the debug information to trace the problem without direct access to the phone.

7.6. Application Summary

In this chapter a system was presented (cf. research question 3) which is not only able to collect and save sensor data but also to extract a variety of meaningful and sophisticated features (cf. research question 1). These features are fed into a classifier which utilizes them to predict the current activities based on previously learned instances. A review process ensures that only high-quality data will be used for training. This is done auto-

[5] `http://developer.android.com/reference/android/app/AlarmManager.html`
[6] `http://crittercism.com`

matically during the night unobserved by the user. Once trained, the application should be able to support the user in tracking daily activities. Results from the classifier are used to preselect and sort the list of potential activities or even detect and store them automatically based on the resulting confidences. The list itself is user-configurable and the selected classifier is very versatile and able to deal with any kind of desired activity.

The modular construction of the application results in a reliable and multifaceted system. While a service in the background is running without the user even noticing it, the user interface is only called upon request. Most of the work, such as recording data, processing them, extracting features, and training the classifier is done automatically in the background. Every major part, like the sensor or the classification system can be configured and extended easily. This enables easy implementation of future enhancements and ideas.

Overall, the application fits the originally requested needs and now has to be evaluated in a study which will show whether it is possible to support users in keeping track of daily activities. This application also saves any kind of raw data (excluding audio) enabling detailed evaluation later on or providing an interesting data set for potential further studies. In the next chapter the application that was built is tested and the effectiveness of the on-board classifier evaluated.

8. Evaluation of the AMARAS Application

A new user study has been conducted to test the developed application. Knowledge gained from the previous study in both the application and in study design was integrated. Yet the new study will only be partially comparable to the last one because of a slightly different focus. This study will finally show whether a smartphone is able to support users in daily activity tracking and if so, how well AMARAS works (cf. research question 4).

8.1. Research Questions for the Study

The first study was all about recording data. The application actually was able to show that it is possible to collect meaningful sensor data with a smartphone. The evaluation of the data then showed that a machine-learning algorithm is able to classify activities from these data. Based on this knowledge the next iteration of the application was built. Aside from the fact that the new application is completely self-contained, the users are now able to choose freely which activities they like to track. This is the difference making it more difficult to compare the two studies directly with each other. The main focus of this study is to examine the following questions:

- Is the developed application able to support the user in activity logging, and are there general improvements in comparison to completely manual digital counterparts?
- How well does a classifier trained with sensor features gained from a smartphone work over time in predicting daily activities of a user with self-defined activities, and how does the classification results behave over time?

Heeding the results from the last study, the classifier will be able to predict some activities very well whereas others still might be problematic. Again this depends very much on the users, but this time they will be able to choose their own activities, hopefully improving the classification results. The classifier gets more training data every day which should make the predictions better with in time. A very interesting point will be the weekend when the users interrupt their normal everyday / working day activities. Weekend activities may be entirely new to the system, which in some likelihood will lead to its failing to guess the activities correctly. As the study is planned to run for 11 days, the second interesting part will begin after the weekend when the classification quality is likely to rise again.

At the end, the goal of this project is to show that an application which runs on a smartphone is able to support users in activity tracking. This study is expected to show this. Supporting means to be more efficient than other traditional or comparable products and the main aspect is to save time. The perfect application would act fully automatically and each step in this direction is a good one. The application might not be able to act completely autonomously but the timesaving aspect is very important and needs to be analyzed. The application is designed to decide if the user has to be queried for the current activities and, if so, the results gained from the classifier are used to preprocess the list of activities. If the system is not completely wrong, this approach will improve the logging process and save time for the users.

In addition to the main focus there are three more peripheral questions which will be inspected:

- How does the self-chosen approach in terms of list of activities differ from the previous study?
- Are the users able to handle the new and much more complex application design?
- Is it theoretically possible to use the application for a longer time period and would the users recommend it?

The study will give an insight as to which degree of granularity and what activities were chosen by the users. The participants will be instructed which kinds of activities are possible to track but in the end everyone can decide individually what is important to her- or himself. Aside from the mere activity logging, several functions are now available for the users and actions such as the review process have to be done properly for good results. The application design aims at reducing the functionality to a minimum but it is still more complex than the previously built application. The users should be able to handle it easily, something that will hopefully be underscored by the questionnaire after the study. Finally, activity logging gets very interesting when done over a long period of time. The study also will show whether a long-term use of AMARAS is satisfactory for the users. With the help of the classifier and the resulting decreased amount of time needed for the process this might be possible.

8.2. Study Conditions

An overview of the conditions for this study can be seen in table 8.1. Because this study is much more time consuming for the users than the previous study, it was possible to recruit only eight subjects. Additionally, such a long lasting study is very expensive and it is hard to find subjects who are willing to operate a system for the entire duration of the study. As mentioned before, the study lasts 11 days including one full weekend. The same smartphones as before were used but this time equipped with new and faster memory cards and SIM cards for mobile internet access. The recording time was reduced to 10 seconds as the main reason for a longer time in the first study was the time the GPS sensors needs to get a valid location fix. This now was implemented with a separate preparation time set to 30 seconds before recording starts. The recording interval and the asking interval is the same as in the first study and the training of the classifier is done at night when the subjects are most likely to be asleep.

The reasoning process of the application is controlled by the two “Auto Confidence” variables. For automatic insertion of activities the individual activity confidence has to be higher than the high value or lower than the low value. If one result is between these values, the user is queried for input. The “Preselect Confidence” value decides if the activity will be preselected or not. The values were derived from pre-studies and already showed good results. The whole reasoning logic is illustrated in figure 8.1. The used classifier is the described MLkNN algorithm with a configuration of 25 neighbors that will

Table 8.1.: Conditions and application settings for the user study.

General Conditions	
Total Number of Subjects	8
Gender Ratio (Male / Female)	6 / 2
Mean Age of Subjects	27
Study Date	October 2012
Study Length	11 Days (incl. 1 Weekend)
Salary for Subjects	40 EUR
Technical Specifications	
Used Smartphones	HTC Desire, Android 2.3.3
Memory Card	SanDisk MicroSDHC Ultra 8GB
SIM Card present	Yes
Internet Connection	Yes (GPRS / EDGE)
Application Version	15 October 2012
Application Settings	
Early Prepare Time	30 Seconds
Prepare Time	10 Seconds
Sensor Recording Time	10 Seconds
Recording Interval	180 Seconds
Ask Interval / Raster	15 Minutes
Training Time	04:01:30 a.m.
Confidence Preselect	0.5
Confidence Auto High	0.8
Confidence Auto Low	0.2
Classifier	MLkNN, Weka
Classifier Settings	25 Neighbors
Pre-Configured Activities	Work, Spare Time, Sleeping, In Company, Transportation, At Home

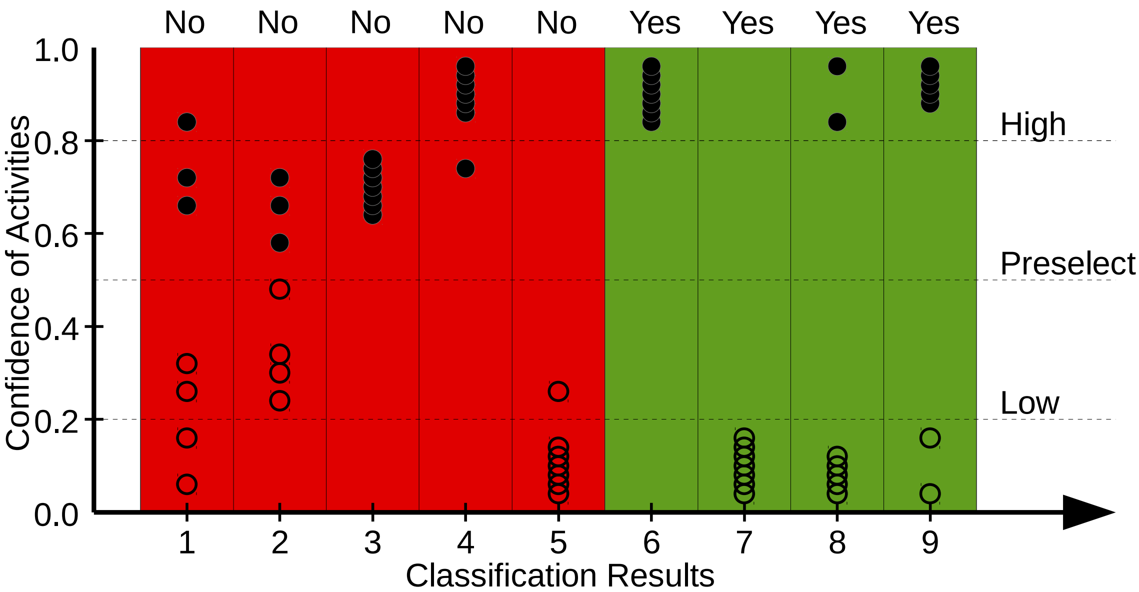


Figure 8.1.: Examples for the automatic insertion of activities based on classifier confidence values. Each activity label confidence (represented by a black circle) has to be higher than the high value of lower than the low value. If all activity labels fulfill this requirement, the set of activities is stored automatically. The first five samples show negative and the last four positive tests. Filled circles are preselected activities based on to the pre-selection threshold.

be used for trainings and predictions. The application comes pre-installed with six very common activities, which can help the user get an idea what is meant by typical activities. Any activity, also the ones given by the application, can be deleted by the users.

Each user got an individual introduction to the application and was given a short printed manual explaining all functions of the application. Before the study started the participants had to fill out a questionnaire. Another one had to be done afterwards. The original documents are in the appendix.

8.2.1. Notes and Problems During the Study

Some users complained about recurring crashes of the application directly after the start of the study. This only has happened occasionally during the two months of testing and was most likely due to the cheap, originally included memory cards. The data attached to the Crittercism crash reports did not help very much in identifying the problem, it was a general "SQLite IOError." Replacing memory cards and / or phones only helped in the case of some users, the crashes still occurred several times a day. After a long search and several tries the bug could be narrowed to a system bug which can happen when writing heavily into the database. For some reasons sometimes the input-output-error occurs and blocks the device for some minutes. The bug could be reproduced with a test application.

As the bug could not be solved easily without a complicated workaround the study had to be and was continued. Three of the eight users were affected at the end and they were

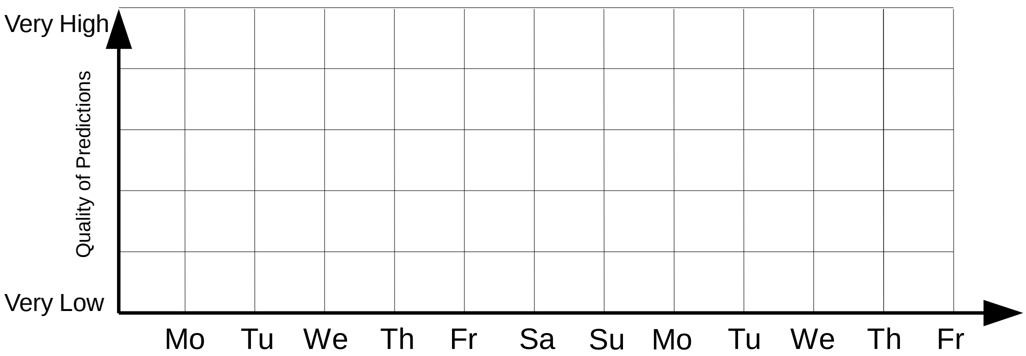


Figure 8.2.: Coordinate system of the after-study questionnaire. The subjects were asked to draw the quality of the predictions over time and mark the days of unusual daily routines.

told to simply restart the application or the entire phone. Control samples have shown that successfully recorded data was not affected and the crashes only led to very small pieces of missing data. Concluding, the main negative effect is the users' experience of the application for those users who had to restart the phone several times a day. The data were still valuable and sufficiently complete to be evaluated in this study. Aside from this, that the study went well and no other problems occurred. Overall, the application was running in a stable and reliable way.

8.3. Questionnaire Results of the Final User Study

The questionnaire that was given to the users before they participated in the study was very similar to the one of the first study. Instead of rating the distribution of the activities they had to write down which activities they would most likely keep track of. The second questionnaire after the study was extended by several statements about how well the AMARAS worked for them on a scale from 0 to 5 with 5 meaning that the participant fully agrees:

- The predictions were good.
- The predictions got better over time.
- The predictions were helpful.
- The predictions sped up the use of the application.
- It is theoretically possible to use this application for a longer time.
- The predictions got worse on the weekend.
- The predictions' quality after the weekend was as high as before.

Additionally, there was a coordinate system into which the users had to draw the development of the predictions' quality subjectively over time as seen in figure 8.2. Also they were asked to mark days that did not match their typical daily routine behavior. This can later be compared to the actual values. The entire questionnaires, in German, are available in the appendix.

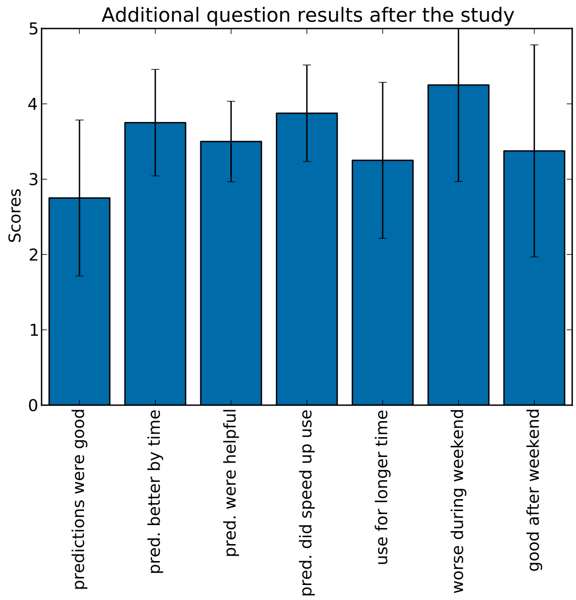


Figure 8.3.: Results of the new, additional statements of the questionnaire completed after the study. The score is cut off at 5 because it is the highest possible score. A high score means that the subject greatly agrees. The bars represent the mean values; the whiskers display the standard deviation.

The first study was very similar for the users in terms of what actions they had to take compared to this study. This is a fact which is reflected in the very similar and comparable results obtained. Therefore, only the newly introduced statements as listed above are interesting here. The results are shown in figure 8.3. While the quality of the predictions depended on the individual subject, they generally agree that the results improved over time. Also the predictions were classified as helpful and the use of the application was sped up. The use of the application for a longer time frame is disputable. As expected, the classifier got subjectively worse on the weekend and recovered for some subjects but not for all.

Overall, the results indicate that the participants felt the classifier had an impact on how the users dealt with AMARAS. Even when suggested activities might not be the best they still helped them and added positive value to their task. The actual performance will be seen later in this chapter when it comes to the detailed evaluation of the classifier results.

8.4. Classification Results Compared to the Last Study

One major technical change compared to the prior study is that the phones now were equipped with SIM cards for mobile internet access. The idea behind this was that WiFi and network localization is available everywhere and, on the one hand, more accurate and, on the other hand, the GPS module was supposed to benefit in terms of a faster localization time. As there were several people who participated in both studies, the localization results can be compared directly by reviewing a day that is similar in its pattern. This was done in figure 8.4. The top row charts were created using data from

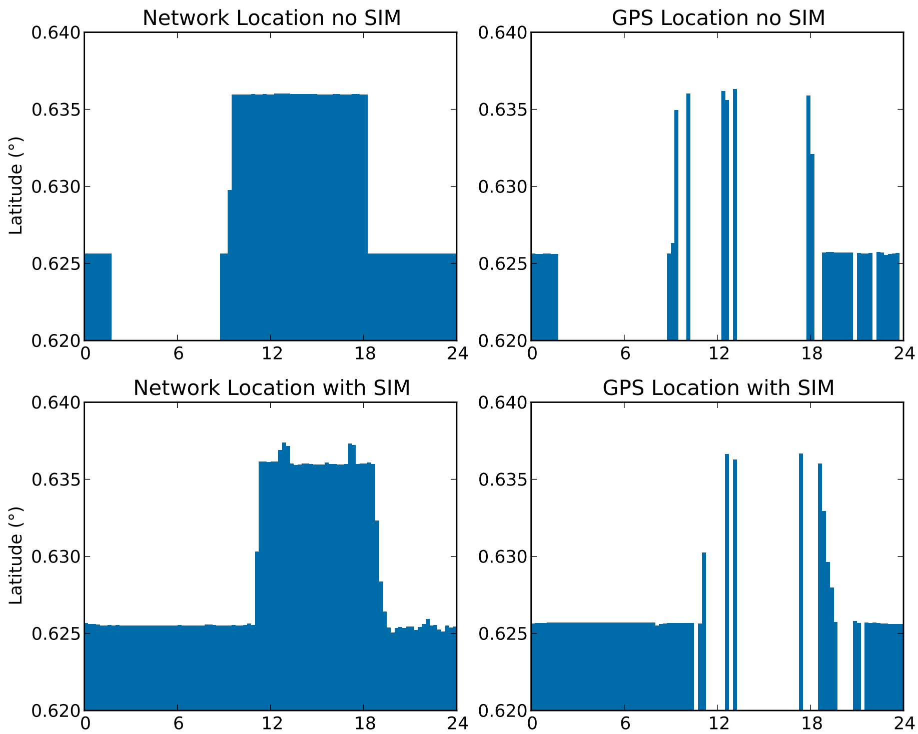


Figure 8.4.: Location information of one comparable day of a subject who participated at both studies. In the first study (top two charts) there was no internet connection available. The network results of the second study (bottom two charts) are much more detailed, especially the network location data.

Table 8.2.: Classification results using the leave-one-out method with a MLkNN algorithm (35 neighbors). All labels were used, including sleep.

User	Act.	H-Loss	Acc.	Prec.	Recall	One-Err
#1	16	0.091	0.392	0.610	0.445	0.117
#2	11	0.089	0.485	0.682	0.553	0.140
#3	13	0.069	0.625	0.762	0.660	0.183
#4	13	0.091	0.381	0.601	0.504	0.175
#5	13	0.088	0.553	0.693	0.604	0.282
#6	13	0.125	0.341	0.494	0.389	0.396
#7	10	0.125	0.449	0.629	0.526	0.263
#8	22	0.052	0.659	0.717	0.695	0.406
Min.	**10**	**0.052**	**0.341**	**0.494**	**0.389**	**0.117**
Max.	**22**	**0.125**	**0.659**	**0.762**	**0.695**	**0.406**
Mean	**13.875**	**0.091**	**0.486**	**0.649**	**0.547**	**0.245**
SD	**3.720**	**0.025**	**0.117**	**0.083**	**0.104**	**0.111**

the first study and the bottom row utilizes the location information from the recent study. The measured latitude value, which was anonymized, is plotted over the hours of one full day starting and ending at midnight.

At first sight it is clearly visible that both days compared are very similar. It is a typical working day of this user and there are two main locations, most likely home and work place. The new plot is no longer missing data during sleep time from 1 a.m. to 8 a.m. as the data logging was done continuously this time. The interesting part for the network location (left column) is that there are more details available in the bottom chart. At the top there are two main latitude values that dominate the whole day. This is because of the fact that the location API method returns the last known value if there is no connection possible to the location server. This results in very vague positioning information. Only when the subject was within the range of a WiFi network, could the actual location be determined; for the rest of the time the last known position was used. The improvements with a working mobile internet connection can be seen clearly at the bottom left chart. At both main locations there are small spikes visible, indicating movements within the building or short breaks in the daily routines. This information was not available before and can be very important for a more detailed analysis of daily activities.

Taking a look at the right column of figure 8.4 the prior assumption of also receiving better GPS location data cannot be confirmed. The phone still had problems locating the subject during work time, assumingly because there is no reception inside the building. Additionally, once a GPS sensor got a fix the next time only a so-called *hot start* or a *warm start* is required utilizing the satellite information from the last fix[1]. Overall, there is no visible improvement but a working mobile internet connection definitely does not worsen the results and can come in handy in some cases, especially when the sensor needs to *cold start*. This can then be compensated by assisted GPS querying an online server.

In order to compare the classifier results to the first study the same leave-one-out method was applied creating the training and test data sets. For these sets the previously intro-

[1] `http://www.gsmarena.com/glossary.php3?term=gps`

Table 8.3.: Classification results using the leave-one-out method with a MLkNN algorithm (35 neighbors). All data labeled with the sleep activity were removed beforehand for better comparability with the prior results.

User	**Act.**	**H-Loss**	**Acc.**	**Prec.**	**Recall**	**One-Err**
#1	16	0.149	0.088	0.410	0.161	0.220
#2	11	0.104	0.477	0.688	0.531	0.158
#3	13	0.105	0.459	0.658	0.521	0.242
#4	13	0.130	0.278	0.504	0.405	0.208
#5	13	0.100	0.475	0.678	0.547	0.292
#6	13	0.166	0.189	0.411	0.252	0.497
#7	10	0.124	0.464	0.597	0.569	0.211
#8	22	0.083	0.169	0.323	0.255	0.627
Min	**10**	**0.083**	**0.088**	**0.323**	**0.161**	**0.158**
Max	**22**	**0.166**	**0.477**	**0.688**	**0.569**	**0.627**
Mean	**13.875**	**0.120**	**0.325**	**0.534**	**0.405**	**0.307**
SD	**3.720**	**0.028**	**0.162**	**0.141**	**0.161**	**0.166**

duced multi-label metrics were calculated using the MLkNN algorithm with 35 neighbors (instead of 25 as used during the study) making them comparable to the first study. The results are shown in table 8.2. Taking a look at these results it can clearly be seen that all mean values are better now than in the previous study. The maximum of the current Hamming Loss values can only be undercut by 19% of the subjects from the first study. 50% of the subjects have an One-Error rate below 0.2 meaning that in 80% of all classifications the label with the highest confidence was correct. In the first study only one subject achieved results this high. The accuracy, precision, and recall values are generally higher while the Hamming Loss value and the One-Error rate are lower. In sum this means that the classifier performs very well and the results are better than the first ones.

When comparing the two studies, there are some additional factors that have to be taken into account. The first one is that there was no “sleep” activity before. This is a special activity which is very easy do detect for a classification algorithm as the conditions while this activity is active are very constant. The phone normally lies still on a flat surface, there are no loud noises around, and the phone is connected to the wall charger as per instruction, another feature considered by the system. Therefore, the sleep label has to be excluded to keep the results comparable. This can be done easily afterwards by modifying the exported instance files that already were computed. Applying this, the evaluation procedure has to be done again and the new, comparable results are shown in table 8.3. As expected, the corrected values are slightly worse than before. Still, even when removing the sleep activity, the results are still better noticeable than the ones of the first study. As the classifier is the same and only the list of features was extended, this may not be the main reason. There is something else missing that definitely has to be taken into account: in the first study the list of activities was predefined and the classified list of activity categories was fixed to 9 items. The second study discards this limit and allowed the subjects to track the activities they like. They were completely free in what and how many activities they wanted to keep track of. For orientation a rough description and some predefined basic activities were handed out to the participants. If unwanted, even the predefined items could be deleted.

No subject used the application to track very few and simple activities only, such as “work” and “spare time” for a work/life balance. The minimum number of activities was 10 and the mean around 14. One subject stands out with 22 activities. For all subjects there was always a mixture of a few very frequent activities accompanied by several others. The typical frequent activity besides sleep is “work,” and an example for a rare one is “lunch” this can be accounted for by the naturally short time frame this activity takes. This distribution is typical and could be seen in the first study, too. The number of activities takes the results on a different level of complexity. On the one hand, it is easier to get low Hamming Loss values as it also depends on the number of activities. But, on the other hand, when considered together with the One-Error rate the true quality of the classifier can be estimated. For example participant #8 has a very low Hamming Loss value, which initially indicates very good classification results. But the large number of labels and the highest One-Error rate of all explain this and, consequently, the overall classification rate is even worse compared to other subjects.

When comparing the two studies with the just discovered correlation in mind, the results of the second study are even better. This may be because of the individualism of activities. While the users had to deal with the predefined activities originally, they now had the chance to adapt the list to their actual daily life. This possibility also has some side effects which are worth noticing: Technically experienced users might adapt their activities to what they think the phone may classify best and will not try out error-prone ones. This of course will improve the results. But on the other hand, people without any technical knowledge about the device and its sensors might do the exact opposite. For the study participants with different background were chosen and the effect was not visible. The customized list of activities enabled the subjects to adapt the classifier to their actual needs without constraints. Overall, the new, advanced feature set combined with the individually customized list of activities led to better classification results than in the first study.

8.5. In-Study Performance Analysis

The multi-label metrics are only a coarse approximation when the leave-one-out method is used. A better approximation for how well AMARAS worked for the users can be reproduced by reviewing the development over the time of the study as illustrated in figure 8.5. For each user for each day of the study training and test sets were created by using the days since the start of the study for training, and the actual day was used as the test set. Walking through the days step by step allows the reproduction of the classifier states the way they were in the study. The metrics were recalculated using these new sets. The mean results for all users separated into days are shown in figure 8.6. The overall classification rate is getting better at the beginning. As expected, the first drawback appears when coming close to the weekend. Most people do not follow their typical daily behavior on weekends which in turn results in very different activities the classifier as a matter of missing information cannot detect. Very interesting are the days after the weekend. When the classifier has really learned about typical daily behavior it should gain similarly good results after the weekend as before it. Exactly this can be seen in the chart in the second week. This indicates that the classifier learned the daily activities

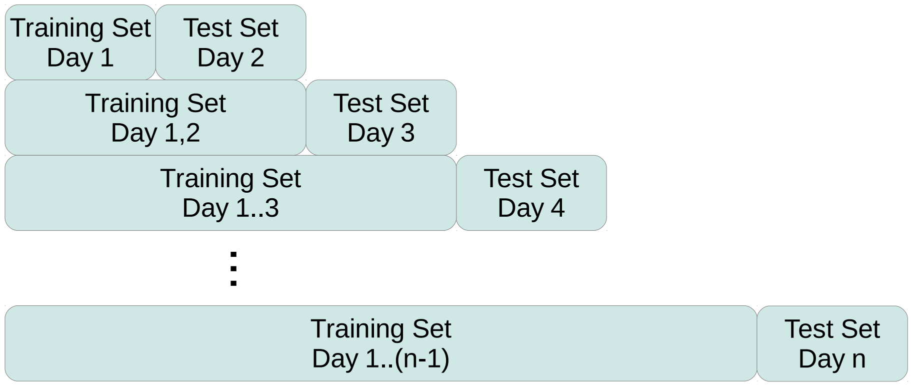


Figure 8.5.: Day-progress method to create test and training data sets by using days before the current day as test data and days before as training data. This reproduces the classifier states as experienced by the subjects during the study.

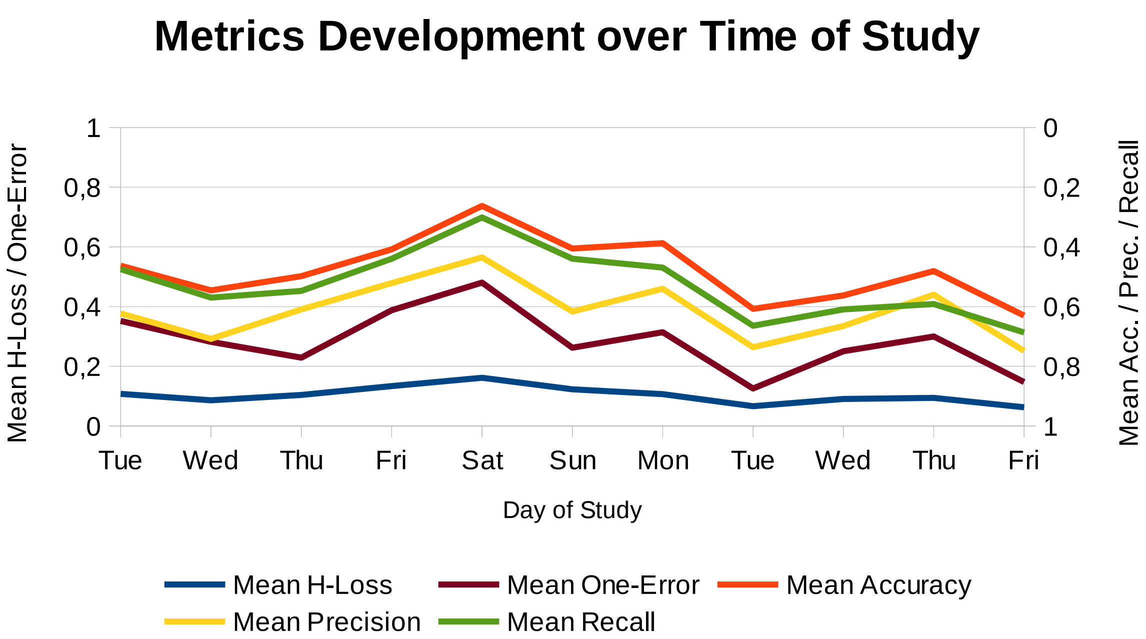


Figure 8.6.: Development of the multi-label metrics over the time of study. The training of the classifier was recreated to match the situation during the study time. The values are the mean values over all subjects by day.

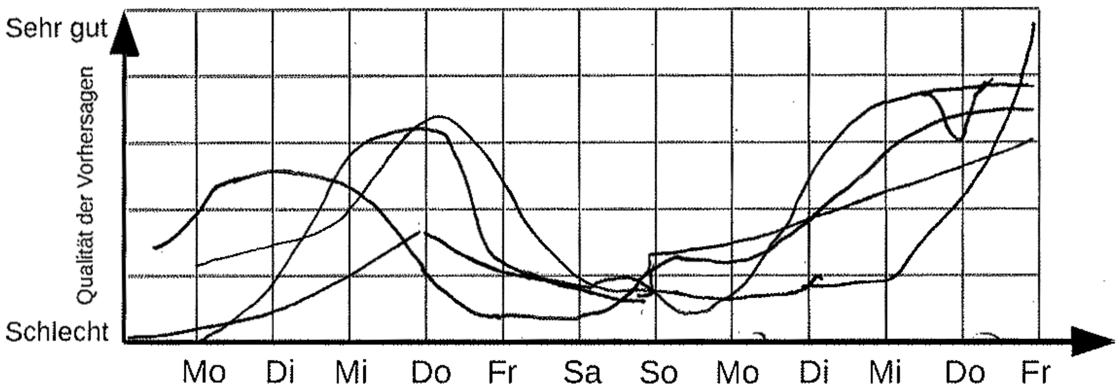


Figure 8.7.: Development of the felt performance of the classifier during the study. A scanned selection of four participants. The y-axis from bottom (bad) to top (very good) displays the subjective quality of the activity label predictions.

and was able to apply this knowledge again after a period of unknown and uncertain input. If there is a general type of behavior of users on weekends, there might be a good chance for the system to learn this with more samples, too. The short time of the study cannot answer this question properly but at least the classification rate on weekdays was reproducible.

One part of the questionnaire after the study was to subjectively rate the quality of the activity label predictions for each day during the study time. These results are very interesting and show how the perceived classification results match the calculated metrics. A selection of the results of four participants is shown in figure 8.7. The remaining participants show similar but not quite as distinctive results. Here the same effect as described previously can be seen: On the weekend the quality was very poor but increased again with the beginning of the second week. The best-felt performance for most users is near the end of the study. Overall, the perceived quality matches the real classification performance well.

For measuring the real effect on how the application had improved daily activity tracking for the subjects, a more concrete metric than the ones presented has to be developed. The original idea of AMARAS is to support the users in keeping track of daily activities and the most useful way of supporting them this is to save time. When comparing different users the metric must be simpler and user independent. A very good and universal approximation is to count changes in the activity list which can be expected to be directly connected to the time the users spend entering activities. Changes are the actions and clicks on the smartphone in order to enter all activities into the system when the suggested list of activities is presented. When using a “dumb” system with no support in any way the users have to enter any single activity manually. The “intelligent” system supports the users by reducing the amount of actions. The optimal case would be a fully automatic system which enters all activities correctly on its own. In order to measure how close a system comes to this perfect one, the number of changes which had to be made can be measured.

Each time AMARAS asks the user to enter activities or the user starts this process manually, there are three possible cases for each activity suggestion:

1. **TruePositive (TP):** The classified activity is preselected and is the one the user wanted to enter into the system. No change required.
2. **FalsePositive (FP):** The classified activity is preselected but the user did not want to enter this one. Therefore, it had to be deselected. One change is required to do this.
3. **FalseNegative (FN):** The activity the users wanted to enter was not preselected by the classifier. The user had to select it manually, one change is required.

The fourth case of true negatives can be neglected as they correlate with the rest and are not used any further. So the number of activities the user originally wanted to enter into the system are the correctly classified ones plus the ones which accidentally were not preselected:

$$AllActivities = TP + FN \tag{8.1}$$

As the false negatives should have been selected in both, manual and wrongly categorized, cases the classifier saved the entries which were true positives minus the one which had to be corrected (false positives) as they were additionally caused by the classifier. This concludes with the new metric which measures the saved changes in percent:

$$SavedChanges = TP - FP \tag{8.2}$$

$$SavedChangesPercent = \frac{SavedChanges}{AllActivities} \tag{8.3}$$

The original confidence value for preselecting an activity was 0.5. To determine the best fitting value for each user the percentage of saved changes for each user can be plotted against the range of pre-selection confidence values as done in figure 8.8. This plot combines two results:

First, there is a black curve for each subject that indicates the percentage of saved changes on the y-axis associated with the confidence pre-selection value on the x-axis. The black vertical line is the original value of 0.5 as used in the study. The highest value achieved for one user here is 52%. The mean value for all subjects is 32% indicated by the broad light blue curve. Secondly, there is the gray circle which stands for the maximum of the mean curve. This spot is the mean best performing confidence value that can be chosen to save a maximum of changes for all subjects. The value is 0.51 and very close to the predefined value used during the study. Overall, there is a large plateau of well-performing confidence values indicating that it is very robust.

For one user the classifier did not support activity tracking during the study at all, but for all the others it had a positive effect. The best performing user saved over half of the activities he entered, which is an excellent result. The fact that the predefined confidence value during the study is very close to the best fitting value for all users, allowed them get an optimal experience of the classifier performance. For even better results this value

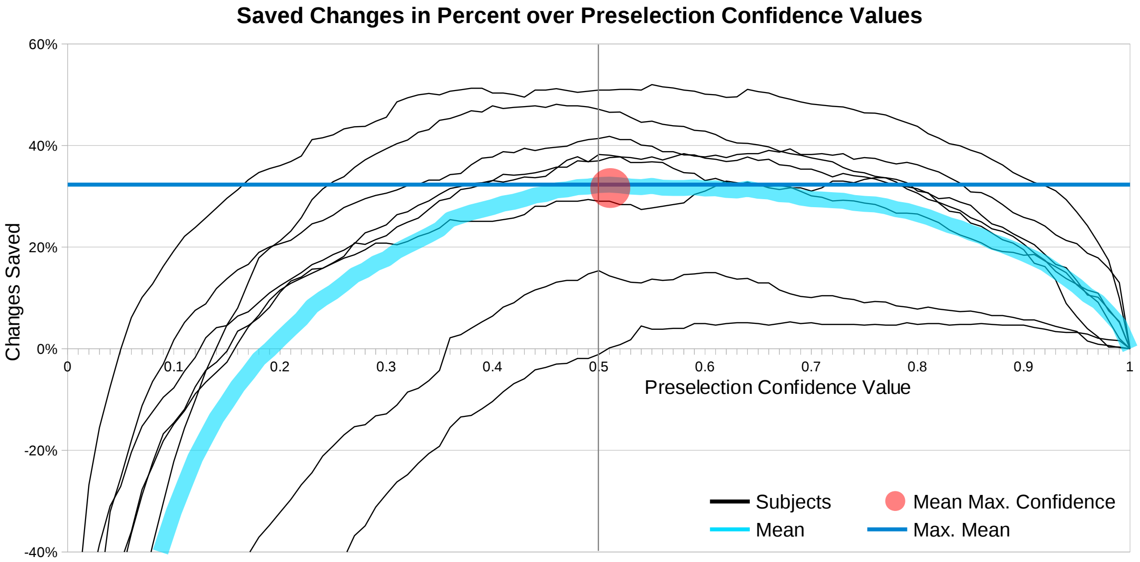


Figure 8.8.: Percentage of saved changes over a range of the reselection confidence values. The black line is the individual subject line. The vertical dark gray line marks the preselection value during the study (0.50) and the gray spot marks the best mean confidence value for all users (0.51).

should have been adapted for each user individually. These results show that AMARAS is definitely able to save time in activity tracking.

8.6. Possible Enhancements

Right now the system is able to check and interrupt for activities in a fixed time interval. One idea to enhance the user experience would be to check for activity changes, and only if a change is detected the system will try to classify the new activities. This would possibly lead to fewer interruptions, e.g. while performing long lasting activities like work and would increase the degree of detail when there are many activity changes in a short period of time. Now the user had to handle this manually by using the silent mode and manual entry. For many users the fixed interval fits very well because of hour-aligned, structured daily processes. But for others a dynamic approach would be much better. If this was not very power intensive, it would be desirable to record and analyze sensor data continuously instead of taking a snapshot every three minutes. Continuous recording would improve performance by providing more data.

Inspecting the classifier and the features used, there might also be room for improvement. First, other classifiers could be tested, and if they perform better, the usability on the phone has to be assured as well. This is also a tradeoff between better results and computation speed on the phone itself. The list of features used is quite long at the moment and it is unclear which features are important and which do not affect the results at all. There also might be features that lower the classification rate and should be removed. Currently, the overall set of features is skewed towards the accelerometer feature which initially is a good idea. As location is also a very important clue for daily activities, this could be represented more. Overall, the selection and weighting of the feature list can definitely be improved.

Some time for the user can be saved with a small modification of the data recording sequences while the application is running: The current version classifies the activities since the last classification time when the activity list screen is started. This can take several seconds. In order to decrease this time, the application could classify the activities every time data are collected. When it comes to the user request, the last results can be taken directly without classifying again because the mean value is used anyway. This reduces the waiting time for the users to a minimum and does not cost extra computing time.

The version used in the study did not use every recorded data for classification. When the system assumed that the device is used at the moment of recording, these data are filtered out assuming no typical behavior for the current activity can be measured. A more sophisticated approach here could also improve the quality of the recorded feature data.

Summing up, there is room left for improvement in all areas. With the knowledge gained from the study the general system process can be enhanced and when investing more time, the classifier and the set of features can be sharpened up, too. There will always be the problem of the limited system and power resources every developer has to deal with. Here, the main improvement for a successor could be to decide how many input sensor data have to be acquired and processed. The preselected values for the study turned out to work very well and fulfilled the requirements completely.

8.7. Summary, Discussion, and Conclusion

After the first study was able to give a very good insight into what kind of sensor data can be collected and how well different classifiers work on extracted feature data, AMARAS incorporates everything learned. It was no longer a recording and labeling application only but the completely newly designed application is completely self-contained. It combines sensor data collection, feature extraction, labeling, and classifier management directly on the device. The goal is to support the user in daily activity logging which is done by learning from past activity entries combined with feature data using it to detect the current situation. The original goal to achieve a fully automatic diary system has changed to a system which supports the users as well as possible in activity tracking. It has to decide if the current activities were detected correctly or if additional user input is necessary. If so, the classifier's confidence results are used to pre-order the activities for a fast and efficient selection by the user. The complete system with all its data is limited to the smartphone and works without any additional hardware or server parts. This ensures a maximum of privacy at all times. The hardware platform is the same as in the first study and theoretically the application should be able to run on any Android device.

A new study was designed in order to survey AMARAS. Similar to the last one, smartphones were handed out to subjects. The task was to use the system as intended by tracking their daily activities. One new aspect is that the users could choose freely the activities they wanted to track. There no longer were predefined activities the users had to make do with.

The goal of this study was to answer the main question as to whether the application is able to support the users in activity tracking. The use of the smartphone was supposed to improve this process. The second goal was to test how AMARAS behaves with the classifier and how well the prediction of activities worked for the users. Additionally, the change over the time of study was interesting to see. Aside from these two main questions, also the difference to the last study and the user interaction with the application was monitored. Eight people participated in the study, which lasted eleven days. Besides some software problems caused by the operating system everything worked well and the data required for a meaningful evaluation could be acquired.

The analysis began with a comparison between the results of the two studies. To get data sets that could well be matched, the sleep activity had to be removed from the current data because it had not been tracked before and improves the results being an easy activity to detect. Utilizing the same metrics slightly better result could be achieved than in the first study. The conditions were reproduced by adapting the training and test sets allowing for a better evaluation of how the system worked for the subjects during the study. The progress over time was rebuilt and showed the behavior of the classifier during the study time. As a final step, a new metric was introduced (cf. research question 4) which is much more meaningful for users: The number of changes to achieve the desired result of logged activities. This is a reliable piece of information on how efficient AMARAS has worked and is comparable to the time the users had to spend dealing with the application. This value depends on the pre-selection confidence threshold and testing this parameter has shown that the initial value used during the study is very close to the best fitting value. Finally, some ideas for future enhancements of the application were proposed.

AMARAS is now completely self-contained and combines all components needed for supported activity tracking in one single smartphone application (cf. research question 3). The standard API compliant software implementation ensures the best possible compatibility to other and future Android devices. The modular design offers a great extensibility for adaptions to other and future projects. New sensors, features, and classifiers can be included and adapted to test and customize the application. The system itself saves the raw data alongside derived feature data and results. This enables detailed and deep data evaluation afterwards. With the raw data new methods can be investigated offline on a much faster computer. This saves times and tasks can be distributed and solved concurrently on compute clusters. Java as the programming language offers a variety of tools and also enables very easy later back porting to the smartphone. The recorded data are a firm basis for ongoing research in many ways. Additionally, the Android ecosystem could be used to distribute the application to a large user base by using the *Android Market Place*. Users all over the world could contribute data by uploading the results to a server. This is only a small next step and can be implemented easily.

The graphical user interface and the application workflow turned out to function very well. There was only little criticism on this. The review process ensures that any entered activity data are double-checked. This is very important as it also ensures that the classifier gets trained with high quality data only. Randomly selected samples have shown that the review process was used to correct wrongly entered activities quite often. Therefore it really was needed because humans naturally make mistakes but also sometimes the users just remembered an activity later. Also, the review is the only chance of correcting entries that were entered fully automatically by the classifier. The very high confidence does not ensure that the items are correct. Especially in completely new situations the predictions are entirely wrong. If the un-reviewed data had been taken for the next training, the classification results would have gotten worse and even might have never detected some activities correctly. The presentation of the assumed list of activities could be improved graphically by small icons belonging to each activity. This would even reduce the time for finding and selection the right item.

The analysis of the data has shown that location information can be heavily improved by supplying a mobile internet connection. As the WiFi or network localization approach does rely on an online service, it worked only in places with a WiFi connection. For the GPS sensor there is no improvement foreseeable because of the short period of time between each localization request. The evaluation of the classifier has shown that the new system behaves even better with the same classifier than in the study before. This can be deduced from the extended list of meaningful features and the individualism of the system. While the users had to deal with fixed activities before, they now had the possibility to customize the list. This customization probably eliminated activities which were hard to detect for the classifier. The results were comparable if not slightly better but taking into consideration that the overall number of tracked activities is higher, it is a great improvement. The classifier had to choose from a larger range of possible activities while making fewer errors. This is a very satisfying result.

In order to recreate the user experience during the study a new method for creating training and test data sets was introduced. By virtually progressing in time the different states of the classifier were restored for each day. The metrics behaved as expected. Up until the weekend the results improved steadily. The weekend was a completely new situation as people do not follow their weekday habits. This, of course, resulted in bad up to completely unusable activity predictions. The most interesting part of the study was after the weekend. When the new week began the error rate went back to its normal rates like in the first week. This showed that the classifier learned the typical weekday activities and is able to detect them. A long-term study could show if there is also a detectable weekend behavior. A new metric was developed for a more realistic measure. The efficiency of AMARAS can be measured based on the changes that had to be done on the activity list to specify the current situation. It is the percentage of saved changed which stands for the quality of the predictions. This measure can also be connected to the individual time each user saved by utilizing the smartphone for activity tracking.

It could be shown that a mean value of 32% of all changes could be saved by the application (cf. research question 5). So about one third of all activities did not to be selected by the user manually. This can not be directly translated to a specific amount of time saved but still shows that it is definitely possible to support the user in activity tracking. AMARAS saves valuable time compared to a naive system where the user has to select each activity manually. For some users even very high savings of over 50% were measured. It is still a long way to a fully automatic system but is was well-known beforehand that this was an unachievable goal. The individual result is heavily influenced by the number of activities the user tracks. The one user for whom AMARAS did not work very well had a very long list of 22 activities while the average user managed only 14 different activities. As a consequence of this long list, there were many labels which were only selected rarely and this ends in bad classification results as the classifier only can predict activities well with a sufficient number of examples. Also, the chances are very high of having many exceptional activities in the list which only appeared once a day and had not been trained before. Overall, the classifier handled common and repeating activities of the users very well.

The final result is that it is possible to build the desired system while respecting all the requirements in terms of an independent, self-contained, extensible, and working application. The results of this study with the presented AMARAS system are a huge improvement and success in supported daily activity tracking.

9. Summary, Conclusion and Outlook

9.1. Summary

Following the idea of an omnipresent recording assistant the goal of this thesis was to develop a system which supports the users in journaling daily activities. A technical system which integrates into the users' lives and acts like a ubiquitous assistant writing everything down the users do. The idea was to get an overview of daily activities and show structures in behavior or extraordinary situations throughout the past. A smartphone was chosen as the technical platform of this project. Many people already own and use smartphones in their everyday lives and without introducing any additional piece of hardware a smartphone-based system is able to unobtrusively accompany the users. The combination of an advanced built-in sensor and hardware technologies was the basis for the planned system. After reviewing current research areas and commercial systems in the area of activity tracking, a detailed evaluation of smartphone operating systems and hardware with special attention on their suitability for such a system was done. Android was selected because it offers prototyping capabilities, and a huge variety of different devices are available.

For a realistic insight into daily activities and corresponding sensor data a first study was conducted where the participants collected and labeled sensor data manually over a period of two weeks by using a smartphone and its built-in sensors. The analysis and evaluation of the recorded data revealed correlations between sensor data and daily activities. Several multi-label classification approaches with machine learning algorithms were tested and showed promising results. To finally show that it is possible to build a mobile, privacy respecting, and supporting system for logging activities, a demonstrator was developed. An application called AMARAS was implemented combining all previous findings and running directly on the smartphone. AMARAS adapts to the users and their individual needs of activity classes. It learns from information it is supplied with from the users and supports them by suggesting, pre-ordering, and pre-selecting current activities up to fully automated logging of daily activities. In order to evaluate the performance of the demonstrator, a new metric was introduced showing that, compared to a manually working system, on average the user is spared 32% up to 52% of all inputs by using AMARAS.

9.2. Conclusion

This thesis was guided by the five research questions stated at the beginning of the work, which can now be discussed and answered:

1) In how far are meaningful features derivable from smartphone sensors in order to classify daily activities?

Many successful research projects in the past have shown that it is possible to use sensors like an accelerometer for detecting specific areas of activities. The analysis of the collected real-life data from smartphones revealed three built-in sensors to be very important and directly connected to specific daily activities: time, location, and acceleration.

The time of day is for many people the most structuring element. Getting up, working hours, or TV shows start and finish at a fixed time and are directly connected to daily activities. Also, some activities can be excluded based on the time of day. In the end the suitability of the time feature highly depends on the user and personal day planning and factors like the user's job. It is very common to have a lot of spare time on weekends and interrupt the structured work days emphasizing again that the time of the week is another very important piece of information.

Aside from time, the location of the user strongly depends on the current activity. This derives from the fact that many buildings and locations are built for specific activities. The office, the own home, the gym, and the home of friends can be detected by the location only and dedicated at least to general activity categories. The coarse location is available with the internal GPS of the phone but inside buildings the detailed location is missing. This would also be interesting as some places or rooms inside have dedicated purposes like the kitchen.

When not staying and being at one place the users typically move between locations. Body movements can be detected by the accelerometer; and different paces and methods of transportation can be differentiated by evaluating these sensors readings. Also types of sports have detectable characteristics, a fact which has been indicated by prior research work.

Overall, the sensors each provide dedicated information about daily activities. The questions was if these pieces of information get lost when all derived features are combined to one large feature vector holding the complete data. AMARAS has shown that a classifier is still able to separate the individual daily activities. How each sensor or combination of sensors is connected to specific daily activities cannot be said reliably, but it would be an interesting follow-up evaluation of the already existing data.

2) In what way do types of daily activities differ in terms of detectability for such a system utilizing internal smartphone sensors only?

As discussed above, all activities that are directly connected to sensor data are relatively easy to detect and can be quickly learned by a classifier. All these activities have in common that the sensor readings represent the activity directly or show individual characteristics during the recording time. These constraints lead to well detectable activities like "Work" or "Spare Time." But also all kinds of transportation differentiate very well. With the pattern centric features for the accelerometer combined with speed information, movement types can be detected very well

It becomes harder for the system to classify activities which are only recognizable through combinations of sensor features and are not directly visible from raw data. One important role is here played by the surrounding sound which may give the key hint for activities which are otherwise not distinguishable. One problem is that sound scenes, such as a crowd of people, are typical for various activities and not always present. In the end, some daily activities can be identified clearly but for others the variety of accumulated features is detrimental.

Whenever an activity does not provide information in one of these areas and is not tied to a daily structure like time, it can generally not be detected. Another main problem is when an activity does not follow any patterns during the recording or manifests itself in many different occurrences, then of course a classifier cannot perform well. Some examples for hard-to-detect activities are “Eating” or “Drinking.” While performing these activities the phone itself is in a very steady situation and no specific behavior can be measured.

In general, the features calculated are able to represent a wide range of activities. Many coarse day structuring events can reliably be detected. Especially short or connecting activities can be problematic but the classification performance still highly depends on the visibility and significance of the sensor values. For some situations dedicated external sensors would help to improve the detectability but the unobtrusiveness of the system must still be guaranteed.

3) What are the key requirements for an automatic system for journaling daily activities running on a smartphone?

The central element of the developed AMARAS system is a classifier which is trained with examples of activities the users enter over time. A separate review process for the activity labels ensures high quality learning data. Once trained, the data are used to compute suggestions of activities and to support the users by minimizing the time needed for input. From the beginning on the application adapts to the users and complies with the demand of this highly individual task of collecting personal daily activities. The individualization was detected as one very important aspect of such a system. Otherwise, users tend to adapt to the system and probably individual structures and patterns of a day cannot be matched and get lost. AMARAS solved this problem by letting the users choose freely which daily activities they want to track.

From the beginning of the application’s implementation the whole development has to be centered around the limited resources of the device. The main factors here are computing power, memory and storage capacity. All methods were selected in such a way that the device can handle the accumulated data and deliver results within an acceptable timespan. Another very important part for mobile systems is battery life. The smartphone has to last at least one full day of normal operating. The computational task of training the classifier is only done at night and when the phone is connected to a wall charger. This approach and the timing of data recording ensures that the user can always operate the device during the day and that it will not shut off because of an empty battery during training, which eventually will corrupt data and leave the system in an uncertain state. The developed software architecture is extendable: it is designed to exchange the classifier and add features or modify the feature calculation. This is useful in order to be able to test configurations without rewriting large parts of the application.

The user interface was designed to minimize the time which is needed to operate the application. When recording activities the input screen is directly called and activities are pre-selected and pre-ordered by the likelihood. In an optimal case the users just have to confirm the selection which only takes seconds. The activity time span is always pre-configured to the time since the last action. For tasks which take a longer time like the review of the entered data an extra review on the large screen of a normal computer was offered. Overall, the user interface is very minimal and clear, small icons support a quick orientation. A user-friendly GUI is crucial to motivate users to actually use the system.

4) What are suitable metrics in order to evaluate an automatic system for journaling daily activities?

Originally, the quality of the gathered data can be compared and evaluated with the help of already developed metrics for classifiers. These metrics were created mainly to compare different machine learning algorithms and give an overview of the performance. Applied to AMARAS, which was built for this thesis, these metrics are not meaningful at the end as they do not provide practical results which reflect the actual usefulness for the users. Therefore, a new metric was developed. The new metric counts the number of required inputs users have to do in order to log their activities manually without any support. This number is compared against the number of inputs which are necessary when suggestions are done by the classifier. This ratio reflects how many inputs are saved by the introduced system.

This metric does not directly correlate to time as users interact differently with AMARAS and have variable learning curves. But it is comparable between users and reflects how well the system worked for them in regard to their individual understanding of daily activities and usage of the application. It can be also used to compare different feature sets and classifiers when working with the same data. For future studies the presets of the configuration can also be optimized.

5) What is the utility of the developed and evaluated system?

In the initial research plan of this project three groups of potential user were introduced. The results of the final study can now be applied to these. Here is again the list of possible users and how each individual group could benefit from the developed and evaluated AMARAS application.

The diarist wants to keep a diary but does not like to spend the amount of time which is normally required to write everything down. By using the application presented here a rough outline of each day could be created in a much smaller amount of time. This outline can be used by the diarist to get an overview of the day and special events can be added to the time line afterwards. The originally long process of writing everything down can be broken down into smaller pieces which may be more comfortable for the user. The form of the resulting diary will definitely differ from traditional diaries but perhaps the collection of typical daily routines combined with personal entries are a new form of diary keeping which especially people like life-loggers take a great interest in.

When medical patients have to do long-term monitoring like an electrocardiography, they have to write down every activity of their day manually. By using a system which is able to guess the current activity or record it automatically this process can be managed more efficiently. The result is a detailed activity diary which enables doctors to diagnose better while keeping the time and effort for the patient at a minimum. In general, when used continuously the system can provide valuable information about patients, which might enable doctors to conclude from past activity entries to current health problems. Also, the system could provide support in reminding patients who suffer from Alzheimer's disease of certain events. The frequent recording of activities could generate an overview of the past which could then assist in the process of recollection. Here, daily structures and outstanding occurrences are be visible and could provide important landmarks in the patients' memory. Summarized, medical patients benefit from a new, efficient way of tracking their daily lives, which helps either themselves or their doctors.

The proposed idea of losing weight by keeping track of meals and drinks may not be the best use scenario for the application developed here. Especially these are the activities which were very hard to detect using the built-in sensors of the smartphones. In general terms the application can be used to determine habits which may not be subjectively noticeable at first sight. Work-life balance, for example, could be easily detected and tracked with the system. Therefore, the application can only partly be recommended as a system which helps users to change their way of living. The system is able to monitor objectively the users' lives and the data can be compared over a long time.

Every application which relies on the detection of activities can make use of the design presented and software implementation. The general approach offers quite a number of operational areas and can easily be adapted to new tasks. Aside from the original purpose, the classifier system could also be used to detect changes in activities and utilizes them for triggering events. Typical behavior of large user groups can be collected and compared to each other. The application offers efficient, digital daily routine tracking which is able to run on a variety of commonly used smartphones.

This thesis has shown that it is possible to derive meaningful features for daily activity tracking from smartphone sensors. The limitation to use built-in sensors limits the range of detectable activities, but in general, day-structuring events and activities can be represented by the available sensor data. External hardware would definitely improve the detectability of daily activities but also comes with additional payload for the users as they would have to deal with extra hardware. This approach was excluded from the beginning on because the basis of the system was to utilize a device the users already own and use to provide a high level of unobtrusiveness. After first tests on standard personal computing hardware the working features and the most promising classification method were successfully ported to the mobile device. AMARAS has shown that it is possible to support users by saving time with a technically unobtrusive device while keeping a diary of daily activities. This was achieved by utilizing a classifier to pre-order and pre-select daily activities. The average saving of 32% of all inputs is still far from a fully automatic system but it definitely reduces the time to maintain a complete list of activities over a longer time. In the end, the time that is needed to record activities might be one key aspect for not keeping an activity diary. In the course of working on this thesis, Android as the smartphone operating system turned out to work very well and can be recommended as an prototyping platform for research projects.

9.3. Outlook

With this project a solid base was developed and a comprehensive collection of data for future work recorded. AMARAS has shown that it is possible to build a flexible, extendable, self-contained system on a smartphone which supports the tracking of daily activities. The biggest room for improvements lies in the selection and calculation of the features combined with a perhaps to be developed and more sophisticated classifier. Although it is very time-intensive, the value of each feature should be determined optimizing the accumulated set of features. The MLkNN classifier which was used is a general purpose classifier. A customized classifier which is optimized for the problem of activity tracking could yet improve the results. Knowledge graphs about activity labels and time series analysis might help here.

In addition to sheer software improvements, another key to more detailed and better results definitely lies in softening the constraint to use the smartphone as the exclusive sensor data source. By connecting the system to the outer world, specialized sensors could deliver meaningful information. One idea is to place small Bluetooth beacons in often used rooms as done by Cheung et al. [21]. These small, low-power devices act passively by providing an ID when the smartphone scans for Bluetooth devices. By limiting the radio range it would be possible to determine the exact location of the user, for example, inside her or his home. This would enable AMARAS to connect activities with exact locations or rooms. A more versatile approach is to locate the smartphone using active Bluetooth beacons as presented by Fischer et al. [32]. Another idea would be to build external triggers which communicate with the smartphone. Imagine a connected water glass or cup which signals the smartphone every time it is used and how much has been drunk [11]. These dedicated triggers would also add very important feature information to the system which cannot be achieved by internal sensors. In general, a smart home could provide a lot of additional information about the user's behavior. Sleeping times, room usage, social contacts and a lot more could be determined by the house system and sent pre-processed to the smartphone.

Aside from all the ideas for external sensory, there is much more valuable information available in a regularly, personally used smartphone. In addition to the sensors, stored data like the calendars, chat or messaging conversations and call logs could be utilized to conclude activities. This is one big area this thesis could not examine because it requires the smartphone to have been used on a regular basis before this study. Also, the general use of a smartphone could give additional valuable information, but to examine all these new possibilities would be another whole research project in itself.

AMARAS was tested to work for at least two weeks. Already at the end of this time period the system behaved very slowly due to the long time the classifier needs to predict activities. It took several seconds waiting time until the result appeared on the screen, but this problem could be solved easily, as already discussed. Training the classifier took about two hours and this time did increase much worse than linearly with time. This could become a problem when the time from the start of training to wake-up gets too short. These are areas where a performance analysis would help to ensure success in long-term operation. A long-term study could also bring better results for comparing the application to already existing, sophisticated methods. Also, it would be interesting to ask traditional diary writers to use AMARAS and analyze their opinion about this new approach. This would also include the extension of the application by the suggested notes and multimedia attachments which would enrich the mere collection of daily activities.

The latest development on the consumer market shows a very strong interest in activity and life tracking. After FitBit presented the "Aria" body scale[1], which is quite similar to the already existing Withings body scale[2], they also announced a new version of their tracking device in form of a wristband called "FitBit Flex"[3] at the CES 2013. Withings counters with an activity tracker of their own called "Withings Smart Activity Tracker"[4], which is analog to the original FitBit device. A new company on the market called

[1] `http://fitbit.com/de/product/aria`
[2] `http://withings.com/de/scales`
[3] `http://engadget.com/2013/01/08/fitbit-flex-hands-on-at-ces-2013/`
[4] `http://withings.com/de/activitytracker/keepmeinformed`

"Fitbug"[5] proves that there is more interest in the activity tracking market and that this is still growing. There are quite a number of devices available for consumers which all allow self-monitoring and the computing of the statistics gained about their daily lives. This trend was started by researchers and some individuals and is now available to everyone. This allows large scale analyses of daily activity data in the future, which is also a very interesting field of research.

In the area of digital diaries Samsung just filed a patent which is all about the detection of daily activities and the generation of stories from the garnered information later presented to the user. The patented idea is very similar to this project and utilizes presented ideas and concepts. The patents description begins with the following:

"An apparatus and method summarize a user's daily life information. The apparatus includes an information collection unit, an analysis unit, a story generator, and a display unit. The information collection unit collects log information including user's daily life information, from at least one electronic device." [6]

With the focus on story-telling the filing is very close to this project. Text passages of the patent like "with the development of wired/wireless technologies, it is made possible to continuously collect information related to a user's daily life" and "[...] there is a need for a technology for summarizing user's life information [...]" are nearly identical to the goals originally proposed in this thesis. The apparatus of this patent consists of "an information collecting unit," "an analysis unit," "a story generator," and "a display unit." At least the first two units are covered by the project of this thesis. The filing of this patent is a very strong indicator for the up-to-dateness and the interest in this area.

This work is only the beginning of a novel approach to collect diary-like data in form of daily activities. The hardware of smartphones will evolve and with new and faster hardware more complex methods are available. Perhaps even continuous data collection will be possible in the future once the battery power problem has been solved. One thing can be determined for sure: The smartphone will get to know its owner better and better. Maybe someday it will know us better than we know ourselves.

[5] `http://engadget.com/2013/01/06/fitbug-bluetooth-orb-wow-activity-sleep-tracker-scales/`
[6] Patent No.: US20120254255 (`http://google.com/patents/US20120254255`)

A. First Study

On the following pages the manual and the questionnaires which were given out to the subjects for the first study are shown.

Anleitung und Hinweise zum Versuch „ActivityTracker“

Kurzanleitung

Das Handy klingelt alle 15 Minuten. Dann bis zu 3 Aktivitäten mit der App auswählen und das Wohlbefinden einstellen. Es ist wichtig, dass das Handy immer an derselben Stelle (am besten eine der vorderen Hosentaschen) am Körper getragen wird. Das Handy muss über Nacht mit dem Aufladegerät aufgeladen werden.

Wenn alles Problemlos funktioniert sollte das Handy alle 3 Minuten angehen und für 20 Sekunden Daten sammeln. Alle 15 Minuten wird man aufgefordert die Aktivitäten einzugeben. Beim Sammeln der Daten wird oben in der Leiste das GPS Symbol eingeblendet.

Allgemeine Hinweise

Aufladen über Nacht

Damit das Handy den Tag über Daten sammeln kann, muss es immer über Nacht aufgeladen werden. In der App vorher die Aktivität „Schlafen“ auswählen und dann das Ladegerät anschließen.

Trageort des Handys

Damit die Daten auswertbar sind, muss das Handy immer an derselben Stelle am Körper getragen werden. Am besten eignet sich dazu eine der vorderen Hosentaschen. Hierbei ist auch die Ausrichtung zu beachten. Das Handy sollte also z.B. immer mit dem Display zum Körper aufrecht getragen werden.

Benutzung des Handys

Während der Versuchszeit darf das Handy zu nichts anderem außer der Aufzeichnungsapp benutzt werden. Dies hat den Grund, dass die aufgenommenen Daten sonst verfälscht werden. Wenn du die normalen Funktionen des Handys ausprobieren möchtest, kannst du es gerne nach der Versuchszeit noch ein paar Tage behalten.

Daten die aufgenommen werden

Das Handy zeichnet während der Versuchszeit alle 3 Minuten für 20 Sekunden Sensordaten auf. Hierzu gehören: Standort, Beschleunigung, Helligkeit, Lautstärke, Bluetoothgeräte, WLANs und die Himmelsrichtung.

Benutzung der App

Klingeltonlautstärke einstellen

Die App meldet sich jede Viertelstunde, immer um voll, 15, 30 und 45 nach, mit einem Klingelton und Vibration. Zusätzlich leuchtet die kleine LED am oberen Bildschirmrand. Mit Hilfe der Lautstärketasten kann die Lautstärke angepasst werden oder das Handy stumm geschaltet werden.

Eingabe der aktuellen Aktivität

Immer wenn das Handy klingelt soll der Benutzer die **Aktivitäten der letzten 15 Minuten eingeben**. Hierzu wird zunächst per Klick die Kategorie ausgewählt und dann die passende

Figure A.1.: Manual for the first study, page 1/2

Aktivität. Trifft keine vorgeschlagene Aktivität zu, kann diese innerhalb der Kategorie unter „Sonstiges“ eingegeben werden.

Sollte man mehr als eine Aktivität in der letzten Viertelstunde ausgeübt haben, **die drei wichtigsten** für sich auswählen und diese nacheinander eingeben.

Zusätzlich zu den Aktivitäten immer noch einmal **das gefühlte Wohlbefinden** mit Hilfe des Sliders unten einstellen. Hat sich dieses gefühlt nicht geändert, einfach kurz darauf tippen damit der Wert gespeichert wird.

Eine Ebene zurück geht es immer mit dem Zurückknopf unten am Handy. Ist man auf der Hauptebene, schließt sich das Programm und wird dann wieder automatisch aufgerufen. Möchte man die App manuell starten, kann man dies mit der Verknüpfung auf dem Desktop tun.

Schlafen und Ruhemodus

Die Optionen „Schlafen“ und „Ruhe“ haben die Besonderheit, dass der viertelstündige Alarm ausgesetzt wird. Nach dem Klick wird man aufgefordert eine Uhrzeit einzugeben, wann die Datenaufnahme und die damit verbunden die Aktivitätsabfrage weiter gehen soll. Die Uhrzeit bezieht sich auf den heutigen Tag, wenn die Uhrzeit an diesem Tag noch erreicht werden kann. Ist die Uhrzeit schon vorbei bezieht sie sich auf den Folgetag.

Fehlerbehebung

Sollte die automatische Erinnerung nicht mehr funktionieren, kann man diese auf der Hauptebene wieder aktivieren. Dazu das Menü mit der Menütaste aufrufen und dort den Punkt „Sammeln neu aktivieren“ auswählen. Danach sollte die App sich wieder alle 15 Minuten melden.

Aktivitäten ansehen und löschen

Es kann vorkommen, dass man aus Versehen eine falsche Aktivität eingibt. Diese kann einfach wieder gelöscht werden. Im Menü (Öffnen mit Menüknopf unter dem Display) gibt es den Punkt „Gespeicherte Aktivitäten anzeigen“. Dies öffnet eine Liste aller Aktivitäten, die bis jetzt eingegeben wurden. Ein Klick auf eine Aktivität löscht diese nach einer Bestätigung.

Eine grobe Auswertung seiner eingegebenen Aktivitäten bekommt man unter dem Punkt „Übersicht über Aktivitäten zeigen“ im Menü.

Sonstiges

Was passiert, wenn der Akku leer ist?

Die App startet sich automatisch nach dem Einschalten des Handys. Es sind keine weiteren Schritte notwendig, wenn das Handy komplett aus war. Dies kann passieren, wenn der Akku nicht lange genug durchhält.

Bei Fragen und Problemen

Ihr könnte mich per E-Mail (shammerl@techfak.uni-bielefeld.de) und per Telefon (0521/10612165) erreichen, wenn ihr Fragen habt, oder es Probleme gibt. Ich bin oft auch in meinem Büro in Q1-138 anzufinden (CITEC Containergebäude).

Figure A.2.: Manual for the first study, page 2/2

Vorfragebogen ActivityTracker

Versuchsperson Nummer:	____________
Geschlecht (einkreisen):	M / W
Alter:	____________

In den nächsten zwei Wochen wirst du ein Handy mit dir tragen und alle 15 Minuten eingeben, welcher Aktivität du nachgegangen bist. In Hinsicht darauf beantworte folgende Fragen:

	Trifft gar nicht zu					Trifft voll zu
Die Unterbrechung wird mich in meinem normalen Tagesablauf beeinträchtigen:	O	O	O	O	O	O
Die Unterbrechung wird mich in meinem normalen Tagesablauf nerven:	O	O	O	O	O	O
Das Benutzen des Systems wird meinen normalen Tagesablauf positiv beeinflussen:	O	O	O	O	O	O
Das Benutzen des Systems wird meinen normalen Tagesablauf negativ beeinflussen:	O	O	O	O	O	O
Ich werde mich an das Benutzen des Systems im Laufe der Zeit gut gewöhnen:	O	O	O	O	O	O
Mein soziales Umfeld wird es stören, dass ich das System benutze:	O	O	O	O	O	O
Mich wird es stören das System in Gegenwart von meinem sozialen Umfeld zu benutzen:	O	O	O	O	O	O

Die Verteilung der Aktivitäten wird vorraussichtlich so aussehen (Schlafen ausgenommen):

Kategorie Essen	____________%
Kategorie Trinken	____________%
Kategorie Hygiene	____________%
Kategorie Arbeit	____________%
Kategorie Freizeit	____________%
Kategorie Haushalt	____________%
Kategorie Soziale Kontakte	____________%
Kategorie Fortbewegung	____________%
Kategorie Sonstiges	____________%

Figure A.3.: Questionnaire before the first study

Nachfragebogen ActivityTracker

Versuchsperson Nummer:	____________
Geschlecht (einkreisen):	M / W
Alter:	____________

Du hast jetzt zwei Wochen ein Handy mit dir getragen und alle 15 Minuten eingeben, welcher Aktivität du nachgegangen bist. In Hinsicht darauf beantworte folgende Fragen:

	Trifft gar nicht zu					Trifft voll zu
Die Unterbrechung haben mich in meinem normalen Tagesablauf beeinträchtigt:	O	O	O	O	O	O
Die Unterbrechung haben mich in meinem normalen Tagesablauf genervt:	O	O	O	O	O	O
Das Benutzen des Systems hat meinen normalen Tagesablauf positiv beeinflusst:	O	O	O	O	O	O
Das Benutzen des Systems hat meinen normalen Tagesablauf negativ beeinflusst:	O	O	O	O	O	O
Ich habe mich an das Benutzen des Systems im Laufe der Zeit gut gewöhnt:	O	O	O	O	O	O
Mein soziales Umfeld hat es gestört, dass ich das System benutze:	O	O	O	O	O	O
Mich hat es gestört das System in Gegenwart von meinem sozialen Umfeld zu benutzen:	O	O	O	O	O	O

Die Verteilung der Aktivitäten hat in etwa so ausgesehen (Schlafen ausgenommen):

Kategorie Essen	____________%
Kategorie Trinken	____________%
Kategorie Hygiene	____________%
Kategorie Arbeit	____________%
Kategorie Freizeit	____________%
Kategorie Haushalt	____________%
Kategorie Soziale Kontakte	____________%
Kategorie Fortbewegung	____________%
Kategorie Sonstiges	____________%

Figure A.4.: Questionnaire after the first study, page 1/3

VP Nr.: _____

Es kam vor, dass ich schon kurz vor dem Alarm auf das Handy geguckt habe:	nie / selten / manchmal / oft / immer
Das Intervall zwischen den Abfragen hätte kürzer sein sollen:	JA / NEIN
Das Intervall zwischen den Abfragen hätte länger sein sollen:	JA / NEIN
Der Abfragezeitpunkt hätte +/- 2 Minuten später/früher sein sollen:	JA / NEIN
Wenn Ja, früher oder später:	Früher / Später
Alle Kategorien die ich in der Zeit gemacht habe waren vorhanden:	JA / NEIN
Wenn Nein: Folgende Kategorien haben gefehlt:	______________________ ______________________ ______________________
Alle Aktivitäten die ich in der Zeit gemacht habe waren vorhanden:	JA / NEIN
Wenn Nein: Folgende Aktivitäten haben gefehlt:	______________________ ______________________ ______________________

	Trifft gar nicht zu					Trifft voll zu
Ich habe mich auf Grund des Systems anders verhalten als normal:	O	O	O	O	O	O
Ich habe mich durch das System überwacht gefühlt:	O	O	O	O	O	O
Ich habe die Benutzung des Systems als Eingriff in meine Privatssphäre empfunden:	O	O	O	O	O	O
Die App könnte meiner Meinung nach helfen um sich zu ändern:	O	O	O	O	O	O
Die App könnte meiner Meinung nach helfen um sich an Vergangenes zu erinnern:	O	O	O	O	O	O
Die App könnte meiner Meinung nach helfen um nicht zu viel zu arbeiten:	O	O	O	O	O	O
Die App könnte meiner Meinung nach helfen um sich besser zu ernähren:	O	O	O	O	O	O
Das Einstellen des Wohlbefindens ist mir leicht gefallen:	O	O	O	O	O	O

Wohlbefinden habe ich wie folgt aufgefasst: ______________________

____

____

Figure A.5.: Questionnaire after the first study, page 2/3

VP Nr.: _____

Ich habe vorher schonmal ein Handy benutzt:	JA	/	NEIN
Ich habe vorher schonmal ein Smartphone benutzt:	JA	/	NEIN
Wenn ja, welches:	______________________		
Ich habe vorher schon einmal ein Android Telefon benutzt:	JA	/	NEIN
Wenn ja, welches:	______________________		

	Trifft gar nicht zu					Trifft voll zu
Die App ließ sich intuitiv bedienen:	O	O	O	O	O	O
Die Bedienung der App war leicht zu verstehen:	O	O	O	O	O	O
Die Bedienung der App war schnell:	O	O	O	O	O	O
Die Bedienung der App was effizient:	O	O	O	O	O	O
Die Bedienung der App hat Spaß gemacht:	O	O	O	O	O	O

Was hätte bei der App besser gemacht werden können: ______________________

____

____

____

____

Allgemeine Anmerkungen: ____

____

____

____

____

____

____

____

Ich stimme der wissenschaftlichen Verwendung und Veröffentlichung der Daten, die durch das Handy und die Fragebögen gesammelt wurden zu: JA / NEIN

______________ ______________

Ort, Datum Unterschrift

Figure A.6.: Questionnaire after the first study, page 3/3

B. Second Study

On the following pages the manual and the questionnaires which were given out to the subjects for the second study are shown.

Anleitung Versuch Cucumber - Aufgabenstellung:

Cucumber heißt die App die auf dem Smartphone läuft. Das Ziel ist es über eure tagtäglichen Aktivitäten Tagebuch zu führen, also was ihr den ganzen Tag so macht und die App soll euch dabei unterstützen. Dazu lernt sie automatisch im Hintergrund wie ihr euch verhaltet und verbindet dieses Wissen mit den von euch eingegebenen Aktivitäten. Aktivitäten werden in ein 15 minütiges Raster eingeteilt. Es ist selbstverständlich möglich, dass ihr mehrere Aktivitäten gleichzeitig ausübt und diese auch mit der App erfasst.

Weil die App anfangs noch nichts über euch weiß wird sie alle 15 Minuten nach euren Aktivitäten seit der letzten Eingabe fragen. Nach ein paar Tagen ist sie in der Lage die Aktivitäten für euch vorzusortieren oder sogar von alleine zu bestimmen welchen Aktivitäten ihr gerade nachgeht. Dann fragt sie auch nicht nach. Diese Informationen nimmt die App aus den verbauten Sensoren. GPS, Bewegungssensoren, etc. liefern die Daten, die für eure Aktivitäten typisch sind und wiedererkannt werden.

Damit bei diesem automatischen Prozess keine Fehler entstehen müssen die erfassten Aktivitäten noch einmal bestätigt werden. Erst dann werden sie vom System zum Lernen verwendet und helfen dabei zukünftige Aktivitäten voraus zu sagen. Damit das System gut arbeiten kann ist eine korrekte Angabe der Aktivitäten notwendig. Gerade in den ersten Tagen prägen die eingegebenen Aktivitäten sehr stark die Voraussagen.

Der Ablauf für euch sieht also folgendermaßen aus:

- Das Smartphone klingelt und ihr werden aufgefordert eure Aktivitäten einzugeben. Dies geschieht in der Regel alle 15 Minuten.
- Falls ihr spontan eure Aktivitäten einfügen wollt, ohne dass das Handy geklingelt hat könnt ihr das durch Drücken von „Aktivitäten einfügen“ tun. Die Zeit wird automatisch so gesetzt, dass sie bis zum letzten Mal reicht.
- Falls die angezeigte zurückliegende Zeit nicht passt kann sie hier noch angepasst werden.
- Neue Aktivitäten können erstellt werden.
- Durch das Bestätigen werden die Aktivitäten gespeichert, sind aber noch nicht überprüft und nachbearbeitet.

Damit das System zuverlässig aus den Daten lernen kann müssen diese noch einmal bestätigt werden. Dies kann zwischendurch, oder einfach am Abend vor dem Schlafen gehen geschehen.

- In der Hauptansicht auf „Review auf dem Handy“ klicken.
- Durch klicken auf einen Zeitslot können Aktivitäten korrigiert werden.
- Grau sind Aktivitäten die den Zeitraum nicht ganz ausgefüllt haben.
- Nach dem Bestätigen werden die Daten gespeichert und werden dann ab dem nächsten Tag mitbenutzt, um Aktivitäten voraus zu sagen.
- „Aktivitäten einfügen“ ermöglicht euch das Einfügen von längeren Aktivitäten.

Figure B.1.: Manual for the second study, page 1/3

An sich ist das auch schon alles. Damit euch der Strom nicht ausgeht, **muss das Handy über Nacht geladen werden**. Außerdem lernt das Handy nachts und muss dafür ausreichend mit Strom versorgt werden. Wenn der Akku nicht für einen vollen Tag ausreichen sollte, müsst ihr ihn, z.B. bei der Arbeit, aufladen.

Weitere Funktionen der App:

- Inkognito Modus: Dieser Modus schützt eure Privatsphäre. Ist er aktiv werden keine Daten aufgenommen und ihr werdet auch nicht nach Aktivitäten gefragt. Der Inkognito Modus wird für einen bestimmten Zeitraum eingeschaltet und deaktiviert sich dann automatisch wieder. Er kann auch manuell, durch erneutes Drücken, deaktiviert werden.
- Ruhemodus: Im Ruhemodus werden im Gegensatz zum Inkognito Modus weiter Daten aufgenommen, ihr werden aber nicht unterbrochen. Dieser Modus ist perfekt für die Schlafenszeit oder Meetings bei denen man ungestört bleiben möchte. Es kann eine Uhrzeit eingestellt werden, ab der es dann wieder normal weiter geht. Nach dem Ruhemodus könnt ihr dann für die gesamte vergangene Zeit Aktivitäten eintragen. Die zurückliegende Zeit wird automatisch eingestellt.
- Review auf dem PC: Neben dem Review direkt auf dem Handy gibt es auch noch die Möglichkeit das bequem an eurem Computer zu erledigen. Hierzu muss das Handy zusammen mit dem Computer zu Hause in eurem Wlan eingebucht sein. Wird der Modus dann gestartet erscheint auf dem Bildschirm eine Adresse, die ihr in euren Browser eingeben müsst. Dort erwartet euch dann eine Kalenderansicht mit den Aktivitäten, die bestätigt werden müssen.
- Löschen von Aktivitäten: Falls ihr euch mit dem Namen vertan habt, oder merkt, dass ihr die Aktivität doch nicht in der Liste braucht, kann sie gelöscht werden. Dafür den Namen gedrückt halten. Anschließend das Löschen bestätigen. Der Vorgang kann mit der Zurück-Taste unten am Handy abgebrochen werden.

Fragen, Antworten und Hinweise:

- Passt euch die Liste der Aktivitäten so an, dass ihr euren Tag damit beschreiben könnt. Dabei darauf achten nicht zu allgemein, aber auch nicht zu detailliert zu sein. Ca. 10-15 Aktivitäten sind optimal.
- Wenn die App euch nicht mehr regelmäßig fragt kann es daran liegen, dass sie Aktivitäten automatisch einträgt. Das könnt ihr nachgucken, indem ihr einfach mal zwischendurch das Review startet. Falls dort keine Aktivitäten vorhanden sind das Telefon einmal neu starten.
- Es kann sein, dass eine Abfrage kommt ob ihr die App beenden oder warten wollt. Hier bitte auf „warten“ drücken. Falls die Abfrage mehrmals kommt hilft es das Handy neu zu starten.
- Sollte die Anzeige „Warte bis die Sensoren die Aufnahme beendet haben...“ mehr als 2 Minuten zu sehen ist, das Handy bitte neu starten.
- Wenn die App beim Start oder zwischendurch mehrmals abstürzt bitte bei mir melden. Es kann dann sein, dass die Speicherkarte defekt ist und ausgetauscht werden muss.

Bei Fragen oder Problemen einfach per E-Mail (shammerl@techfak.uni-bielefeld.de) oder per Telefon (Büro: 0521 / 106 12165) bei mir melden.

Figure B.2.: Manual for the second study, page 2/3

Kurzanleitung:

- Alle 15 Minuten, oder immer wenn das Handy euch benachrichtigt die Aktivitäten seit dem letzten Mal eingeben. Mehrere Aktivitäten sind möglich und es kann auch spontan eingefügt werden.
- **Abends das Review starten** um die Daten zu verifizieren. Anschließend zum Schlafen in den Ruhemodus bis zum nächsten Morgen setzen. **Handy an das Ladekabel anschließen!**
- Falls die App nicht mehr bedienbar ist, oder am Abend keine Aktivitäten zum Verifizieren vorhanden sind das Handy einfach neu starten. Danach sollte alles wieder anfangen zu arbeiten.

Aktivitäten:

- Aktivitäten sind Dinge die ihr macht oder ein den Zustand beschreibt den ihr macht. Ihr könnt gerade Arbeiten oder eure Freizeit genießen. Sport machen oder Fernsehen. Mit Freunden zusammen sein oder alleine am Computer arbeiten. Am besten funktioniert das System mit Aktivitäten bei denen ihr ein bestimmtes Verhalten aufweist, also euch an einem bestimmten Ort aufhaltet oder ein bestimmtes Bewegungsmuster durchführt.
- Welche Aktivitäten ihr festlegt ist euch überlassen und ist auch bei jedem Menschen anders. Am Anfang sind ein paar Beispiele eingetragen. Fügt einfach die Aktivitäten hinzu die ihr braucht. Am besten macht ihr euch am Anfang einmal Gedanken welche Aktivitäten in eurem Tagebuch sein sollen. Es ist aber auch kein Problem die Liste später zu erweitern.

Aufgenommene Daten:

- Damit die Aktivitäten bestimmt werden können, sammelt das Handy Daten aus den verbauten Sensoren. Hier ist eine Liste aller aufgenommenen und gespeicherten Daten, die während der Versuchszeit anfallen:
  - Datum und Zeit
  - Zustand des Handys (Batterie, Bildschirm an/aus) und wann die App benutzt wird
  - Ort an dem ihr euch aufhaltet und die Ausrichtung
  - Beschleunigungswerte
  - Merkmale der Umgebungsgeräusche. Es gibt keinen Tonmitschnitt!
  - Verfügbare WLans und Bluetooth Geräte in der Umgebung
  - Lichtintensität und Entfernung vom Handy (reicht nur einige cm)
  - Eingegebene Aktivitäten. Gelöschte Aktivitäten werden zur Auswertung aufgehoben!

Bei Fragen oder Problemen einfach per E-Mail (shammerl@techfak.uni-bielefeld.de) oder per Telefon (Büro: 0521 / 106 12165) bei mir melden.

Figure B.3.: Manual for the second study, page 3/3

Vorfragebogen Cucumber

Versuchsperson Nummer: ____________

Geschlecht (einkreisen): M / W

Alter: ____________

In den nächsten zwei Wochen wirst du ein Handy mit dir tragen und alle 15 Minuten eingeben, welcher Aktivität du nachgegangen bist. In Hinsicht darauf beantworte folgende Fragen:

	Trifft gar nicht zu					Trifft voll zu
Die Unterbrechung wird mich in meinem normalen Tagesablauf beeinträchtigen:	O	O	O	O	O	O
Die Unterbrechung wird mich in meinem normalen Tagesablauf nerven:	O	O	O	O	O	O
Das Benutzen des Systems wird meinen normalen Tagesablauf positiv beeinflussen:	O	O	O	O	O	O
Das Benutzen des Systems wird meinen normalen Tagesablauf negativ beeinflussen:	O	O	O	O	O	O
Ich werde mich an das Benutzen des Systems im Laufe der Zeit gut gewöhnen:	O	O	O	O	O	O
Mein soziales Umfeld wird es stören, dass ich das System benutze:	O	O	O	O	O	O
Mich wird es stören das System in Gegenwart von meinem sozialen Umfeld zu benutzen:	O	O	O	O	O	O

Ich habe schon an der ersten Studie teilgenommen: JA / NEIN

Überlege kurz, welche Aktivitäten du in der Zeit der Studie aufzeichnen möchtest und schreibe sie hier in. Du bist nachher nicht auf diese Aktivitäten beschränkt, es geht nur um eine grobe Einschätzung.

Aktivitäten sind z.B.: Freizeit / Arbeit / In Gesellschaft / Sport / Schlafen / Auto fahren / Zu Fuß gehen / …

____

____

____

Figure B.4.: Questionnaire before the second study

Nachfragebogen Cucumber

Versuchsperson Nummer:	____________
Geschlecht (einkreisen):	M / W
Alter:	____________

Du hast jetzt zwei Wochen ein Handy mit dir getragen und alle 15 Minuten eingeben, welcher Aktivität du nachgegangen bist. In Hinsicht darauf beantworte folgende Fragen:

	Trifft gar nicht zu					Trifft voll zu
Die Unterbrechung haben mich in meinem normalen Tagesablauf beeinträchtigt:	O	O	O	O	O	O
Die Unterbrechung haben mich in meinem normalen Tagesablauf genervt:	O	O	O	O	O	O
Das Benutzen des Systems hat meinen normalen Tagesablauf positiv beeinflusst:	O	O	O	O	O	O
Das Benutzen des Systems hat meinen normalen Tagesablauf negativ beeinflusst:	O	O	O	O	O	O
Ich habe mich an das Benutzen des Systems im Laufe der Zeit gut gewöhnt:	O	O	O	O	O	O
Mein soziales Umfeld hat es gestört, dass ich das System benutze:	O	O	O	O	O	O
Mich hat es gestört das System in Gegenwart von meinem sozialen Umfeld zu benutzen:	O	O	O	O	O	O

Es kam vor, dass ich schon kurz vor dem Alarm auf das Handy geguckt habe:	nie / selten / manchmal / oft / immer
Das Intervall zwischen den Abfragen hätte kürzer sein sollen:	JA / NEIN
Das Intervall zwischen den Abfragen hätte länger sein sollen:	JA / NEIN

	Trifft gar nicht zu					Trifft voll zu
Ich habe mich auf Grund des Systems anders verhalten als normal:	O	O	O	O	O	O
Ich habe mich durch das System überwacht gefühlt:	O	O	O	O	O	O
Ich habe die Benutzung des Systems als Eingriff in meine Privatsphäre empfunden:	O	O	O	O	O	O
Die App könnte meiner Meinung nach helfen um sich zu ändern:	O	O	O	O	O	O

Figure B.5.: Questionnaire after the second study, page 1/3

Die App könnte meiner Meinung nach helfen um sich an Vergangenes zu erinnern:	O	O	O	O	O	O
Die App könnte meiner Meinung nach helfen um nicht zu viel zu arbeiten:	O	O	O	O	O	O
Die App könnte meiner Meinung nach helfen um sich besser zu ernähren:	O	O	O	O	O	O

Welche Aktivitäten hast du zu dem System hinzugefügt und was bedeuteten diese für dich:

____

____

____

Nenne jeweils zwei Beispiele für Aktivitäten, die das System gut / schlecht erkannt hat:

Gut erkannt: ____

Schlecht erkannt: ____

	Trifft gar nicht zu					Trifft voll zu
Die Vorhersagen waren gut:	O	O	O	O	O	O
Mit der Zeit wurden die Vorhersagen besser:	O	O	O	O	O	O
Die Vorhersagen waren hilfreich:	O	O	O	O	O	O
Die Vorhersagen haben die Benutzung schneller gemacht:	O	O	O	O	O	O
Es ist theoretisch möglich, die App über einen längeren Zeitraum zu benutzen (Monate / Jahre):	O	O	O	O	O	O
Die Vorhersagen waren am Wochende schlechter	O	O	O	O	O	O
Die Vorhersagen waren nach dem Wochenende wieder so gut wie vor dem Wochenende	O	O	O	O	O	O

Zeichne über den Versuchzeitraum über hinweg die gefühlte Qualität der Vorhersagen auf. Kreise die Tage ein, an denen kein „normaler“ Tagesablauf vorhanden war.

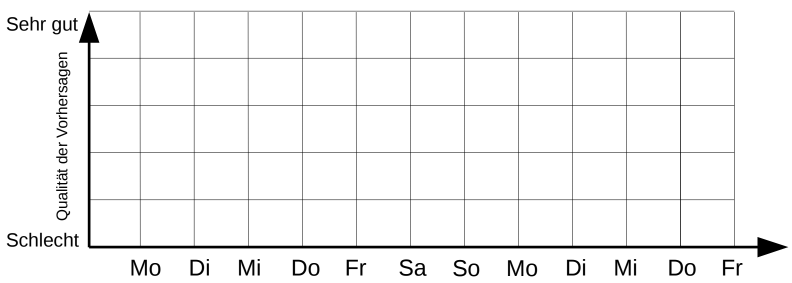


Figure B.6.: Questionnaire after the second study, page 2/3

Ich habe vorher schon mal ein Handy benutzt:	JA	/	NEIN
Ich habe vorher schon mal ein Smartphone benutzt:	JA	/	NEIN

Wenn ja, welches: ______________________________

Ich habe vorher schon einmal ein Android Telefon benutzt:	JA	/	NEIN

Wenn ja, welches: ______________________________

	Trifft gar nicht zu					Trifft voll zu
Die App ließ sich intuitiv bedienen:	O	O	O	O	O	O
Die Bedienung der App war leicht zu verstehen:	O	O	O	O	O	O
Die Bedienung der App war schnell:	O	O	O	O	O	O
Die Bedienung der App war effizient:	O	O	O	O	O	O
Die Bedienung der App hat Spaß gemacht:	O	O	O	O	O	O

Was hätte bei der App besser gemacht werden können: ______________________________

____

____

Allgemeine Anmerkungen: ______________________________

____

____

____

Ich stimme der wissenschaftlichen Verwendung und Veröffentlichung der Daten, die durch das Handy und die Fragebögen gesammelt wurden zu:	JA	/	NEIN

______________________ ______________________
Ort, Datum Unterschrift

Figure B.7.: Questionnaire after the second study, page 3/3

Bibliography

[1] K. Aizawa. “Digitizing Personal Experiences: Capture and Retrieval of Life Log”. In: *Proceedings of the 11th International Multimedia Modelling Conference, 2005*. IEEE. 2005, pages 10–15.

[2] K. Aizawa, T. Hori, S. Kawasaki, and T. Ishikawa. “Capture and Efficient Retrieval of Life Log”. In: *Proceedings of the Pervasive 2004 Workshop on Memory and Sharing of Experiences, Linz/Vienna, Austria*. 2004.

[3] K. Aizawa, D. Tancharoen, S. Kawasaki, and T. Yamasaki. “Efficient Retrieval of Life Log Based on Context and Content”. In: *Proceedings of the the 1st ACM workshop on Continuous archival and retrieval of personal experiences*. ACM. 2004, pages 22–31.

[4] M. Azizyan, I. Constandache, and R.R. Choudhury. “SurroundSense: Mobile Phone Localization via Ambience Fingerprinting”. In: *Proceedings of the 15th annual international conference on Mobile computing and networking*. ACM. 2009, pages 261–272.

[5] P. Bahl and V.N. Padmanabhan. “RADAR: An In-Building RF-Based User Location and Tracking System”. In: *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*. Volume 2. IEEE. 2000, pages 775–784.

[6] L. Bao and S. Intille. “Activity Recognition from User-Annotated Acceleration Data”. In: *Pervasive Computing* 3001 (2004), pages 1–17.

[7] G. Bell. “A Personal Digital Store”. In: *Communications of the ACM* 44.1 (2001), pages 86–91.

[8] G. Bell and J. Gemmell. “A Digital Life”. In: *Scientific American Magazine* 296.3 (2007), pages 58–65.

[9] S.H.I. Bell and T.C. Skeat. *The New Gospel Fragments*. Trustees of the British Museum, 1935.

[10] F. Ben Abdesslem, A. Phillips, and T. Henderson. “Less is More: Energy-Efficient Mobile Sensing with SenseLess”. In: *Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds*. ACM. 2009, pages 61–62.

[11] R. Bhattacharyya, C. Floerkemeier, and S. Sarma. “RFID Tag Antenna Based Sensing: Does Your Beverage Glass Need a Refill?” In: *2010 IEEE International Conference on RFID*. IEEE. 2010, pages 126–133.

[12] P.E. Black. *Dictionary of Algorithms and Data Structures*. National Institute of Standards and Technology, 1998.

[13] T. Brezmes, J.L. Gorricho, and J. Cotrina. “Activity Recognition from Accelerometer Data on a Mobile Phone”. In: *Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living* 5518 (2009), pages 796–799.

[14] M.K. Buckland. “Emanuel Goldberg, Electronic Document Retrieval, and Vannevar Bush’s Memex”. In: *Journal of the American Society for Information Science* 43.4 (1992), pages 284–294.

[15] Statistisches Bundesamt. *Statistisches Jahrbuch für die Bundesrepublik Deutschland 2011*. Statistisches Bundesamt, 2011.

[16] V. Bush. “As We May Think”. In: *Atlantic Monthly* 176.1 (1945), pages 101–108.

[17] J.B.J. Bussmann, W.L.J. Martens, J.H.M. Tulen, FC Schasfoort, H.J.G. van den Berg-Emons, and HJ Stam. “Measuring Daily Behavior using Ambulatory Accelerometry: The Activity Monitor”. In: *Behavior Research Methods* 33.3 (2001), pages 349–356.

[18] A. Carroll and G. Heiser. “An Analysis of Power Consumption in a Smartphone”. In: *Proceedings of the 2010 USENIX Annual Technical Conference*. USENIX Association. 2010, pages 270–284.

[19] J. Chaudhari, S.S. Cheung, and M.V. Venkatesh. “Privacy Protection for Life-Log Video”. In: *IEEE Workshop on Signal Processing Applications for Public Security and Forensics, 2007. SAFE’07.* IEEE. 2007, pages 1–5.

[20] M. Chen, T. Sohn, D. Chmelev, D. Haehnel, J. Hightower, J. Hughes, A. LaMarca, F. Potter, I. Smith, and A. Varshavsky. “Practical Metropolitan-Scale Positioning for GSM Phones”. In: *UbiComp 2006: Ubiquitous Computing* 4206 (2006), pages 225–242.

[21] K.C. Cheung, S.S. Intille, and K. Larson. “An Inexpensive Bluetooth-Based Indoor Positioning Hack”. In: *Proc. UbiComp06 Extended Abstracts* (2006).

[22] D.J. Cook and S.K. Das. *Smart Environments: Technologies, Protocols, and Applications*. Volume 43. Wiley-Interscience, 2005.

[23] T. Cover and P. Hart. “Nearest Neighbor Pattern Classification”. In: *IEEE Transactions on Information Theory* 13.1 (1967), pages 21–27.

[24] Android Developers. *Platform Versions*. CTAN: `http://developer.android.com/resources/dashboard/platform-versions.html`. Feb. 2012.

[25] A.R. Doherty and A.F. Smeaton. “Combining Face Detection and Novelty to Identify Important Events in a Visual Lifelog”. In: *IEEE 8th International Conference on Computer and Information Technology Workshops, 2008. CIT Workshops 2008.* IEEE. 2008, pages 348–353.

[26] A.R. Doherty, A.F. Smeaton, K. Lee, and D.P.W. Ellis. “Multimodal Segmentation of Lifelog Data”. In: *Large Scale Semantic Access to Content (Text, Image, Video, and Sound)*. Le Centre De Hautes Etudes Internationales D’Informatique Documentaire. 2007, pages 21–38.

[27] N. Eagle and A. Pentland. “Reality Mining: Sensing Complex Social Systems”. In: *Personal and Ubiquitous Computing* 10.4 (2006), pages 255–268.

[28] N. Eagle and A.S. Pentland. “Eigenbehaviors: Identifying Structure in Routine”. In: *Behavioral Ecology and Sociobiology* 63.7 (2009), pages 1057–1066.

[29] K. Farrahi and D. Gatica-Perez. “Daily Routine Classification from Mobile Phone Data”. In: *Machine Learning for Multimodal Interaction* 5237 (2008), pages 173–184.

[30] K. Farrahi and D. Gatica-Perez. “What Did You Do Today?: Discovering Daily Routines From Large-Scale Mobile Data”. In: *Proceedings of the 16th ACM international conference on Multimedia*. ACM. 2008, pages 849–852.

[31] K. Farrahi and D. Gatica-Perez. “Learning and Predicting Multimodal Daily Life Patterns from Cell Phones”. In: *Proceedings of the 2009 international conference on Multimodal interfaces*. ACM. 2009, pages 277–280.

[32] G. Fischer, B. Dietrich, and F. Winkler. “Bluetooth Indoor Localization System”. In: *Proceeding Workshop on Positioning, Navigation and Communication*. 2004, pages 147–56.

[33] J. Gemmell, G. Bell, R. Lueder, S. Drucker, and C. Wong. “MyLifeBits: Fulfilling the Memex Vision”. In: *Proceedings of the tenth ACM international conference on Multimedia*. ACM. 2002, pages 235–238.

[34] J. Gemmell, G. Bell, and R. Lueder. “MyLifeBits: A Personal Database for Everything”. In: *Communications of the ACM* 49.1 (2006), pages 88–95.

[35] Shantanu Godbole and Sunita Sarawagi. “Discriminative Methods for Multi-labeled Classification”. In: *Advances in Knowledge Discovery and Data Mining*. Volume 3056. Springer, 2004, pages 22–30.

[36] E. Goldberg. “The Retrieval Problem in Photography(1932)”. In: *Journal of the American Society for Information Science* 43.4 (1992), pages 295–298.

[37] A. Haeberlen, E. Flannery, A.M. Ladd, A. Rudys, D.S. Wallach, and L.E. Kavraki. “Practical Robust Localization over Large-Scale 802.11 Wireless Networks”. In: *Proceedings of the 10th annual international conference on Mobile computing and networking*. ACM. 2004, pages 70–84.

[38] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten. “The WEKA Data Mining Software: An Update”. In: *ACM SIGKDD Explorations Newsletter* 11.1 (2009), pages 10–18.

[39] J. Healey and R.W. Picard. “StartleCam: A Cybernetic Wearable Camera”. In: *Second International Symposium on Wearable Computers, 1998*. IEEE. 1998, pages 42–49.

[40] T. Heiberg, T.K. Kvien, Ø. Dale, P. Mowinckel, G.J. Aanerud, A.B. Songe-Møller, T. Uhlig, and K.B. Hagen. “Daily Health Status Registration (Patient Diary) in Patients with Rheumatoid Arthritis: A Comparison Between Personal Digital Assistant and Paper-Pencil Format”. In: *Arthritis Care & Research* 57.3 (2007), pages 454–460.

[41] S. Hodges, L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur, and K. Wood. “SenseCam: A Retrospective Memory Aid”. In: *UbiComp 2006: Ubiquitous Computing* (2006), pages 177–193.

[42] T. Hori and K. Aizawa. "Context-Based Video Retrieval System for the Life-Log Applications". In: *Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval.* ACM. 2003, pages 31–38.

[43] T. Iso and K. Yamazaki. "Gait Analyzer based on a Cell Phone with a Single Three-Axis Accelerometer". In: *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services.* ACM. 2006, pages 141–144.

[44] S. Katz. "Assessing Self-Maintenance: Activities of Daily Living, Mobility, and Instrumental Activities of Daily Living". In: *Journal of the American Geriatrics Society* 31.12 (1983), pages 721–727.

[45] S. Katz and C.A. Akpom. "A Measure of Primary Sociobiological Functions". In: *International Journal of Health Services* 6.3 (1976), pages 493–508.

[46] S. Katz and C.A. Akpom. *Katz Basic Activities of Daily Living (ADL) Scale @ONLINE.* 1976. URL: `http://son.uth.tmc.edu/coa/FDGN_1/RESOURCES/ADLandIADL.pdf`.

[47] S. Katz, T.D. Downs, H.R. Cash, and R.C. Grotz. "Progress in Development of the Index of ADL". In: *The Gerontologist* 10.1 Part 1 (1970), pages 20–30.

[48] N. Kiukkonen, J. Blom, O. Dousse, D. Gatica-Perez, and J. Laurila. "Towards Rich Mobile Phone Datasets: Lausanne Data Collection Campaign". In: *Proc. ICPS, Berlin* (2010).

[49] H. Kwak, C. Lee, H. Park, and S. Moon. "What is Twitter, a Social Network or a News Media?" In: *Proceedings of the 19th international conference on World wide web.* ACM. 2010, pages 591–600.

[50] J.R. Kwapisz, G.M. Weiss, and S.A. Moore. "Activity Recognition Using Cell Phone Accelerometers". In: *ACM SIGKDD Explorations Newsletter* 12.2 (2011), pages 74–82.

[51] A.M. Ladd, K.E. Bekris, A.P. Rudys, D.S. Wallach, and L.E. Kavraki. "On the Feasibility of Using Wireless Ethernet for Indoor Localization". In: *IEEE Transactions on Robotics and Automation* 20.3 (2004), pages 555–559.

[52] A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, et al. "Place Lab: Device Positioning Using Radio Beacons in the Wild". In: *Pervasive Computing* 3468 (2005), pages 116–133.

[53] M.P. Lawton and E.M. Brody. "Assessment of Older People: Self-Maintaining and Instrumental Activities of Daily Living". In: *The Gerontologist* (1969).

[54] H. Lee, A.F. Smeaton, N.E. O'Connor, G. Jones, M. Blighe, D. Byrne, A. Doherty, and C. Gurrin. "Constructing a SenseCam Visual Diary as a Media Process". In: *Multimedia Systems* 14.6 (2008), pages 341–349.

[55] M.L. Lee and A.K. Dey. "Providing Good Memory Cues for People with Episodic Memory Impairment". In: *Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility.* ACM. 2007, pages 131–138.

[56] M.L. Lee and A.K. Dey. "Lifelogging Memory Appliance for People with Episodic Memory Impairment". In: *Proceedings of the 10th international conference on Ubiquitous computing.* ACM. 2008, pages 44–53.

[57] Christian Leichsenring. “Subliminal Copresence Systems”. PhD thesis. Bielefeld University, 2013. unpublished.

[58] L. Liao, D. Fox, and H. Kautz. “Extracting Places and Activities from GPS Traces Using Hierarchical Conditional Random Fields”. In: *The International Journal of Robotics Research* 26.1 (2007), pages 119–134.

[59] B. Logan et al. “Mel Frequency Cepstral Coefficients for Music Modeling”. In: *International Symposium on Music Information Retrieval*. Volume 28. 2000, page 5.

[60] M. Makikawa, S. Kurata, Y. Higa, Y. Araki, and R. Tokue. “Ambulatory Monitoring of Behavior in Daily Life by Accelerometers Set at Both-Near-Sides of the Joint”. In: *Studies in health technology and informatics* 84.1 (2001), pages 840–843.

[61] MJ Mathie, A.C.F. Coster, NH Lovell, and BG Celler. “Detection of Daily Physical Activities using a Triaxial Accelerometer”. In: *Medical and Biological Engineering and Computing* 41.3 (2003), pages 296–301.

[62] Y. Mathpal. *Prehistoric Rock Paintings of Bhimbetka*. Abhinav Publications, 1984.

[63] G.A. Miller. “The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity For Processing Information”. In: *The Psychological Review* 63.2 (1956), page 81.

[64] H. Mizuno, K. Sasaki, and H. Hosaka. “Indoor-Outdoor Positioning and Lifelog Experiment with Mobile Phones”. In: *Proceedings of the 2007 workshop on Multimodal interfaces in semantic interaction*. ACM. 2007, pages 55–57.

[65] B.A. Nardi, D.J. Schiano, and M. Gumbrecht. “Blogging as Social Activity, or, Would You Let 900 Million People Read Your Diary?” In: *Proceedings of the 2004 ACM conference on Computer supported cooperative work*. ACM. 2004, pages 222–231.

[66] B. Nham, K. Siangliulue, and S. Yeung. *Predicting Mode of Transport from iPhone Accelerometer Data*. Technical report. Technical report, Stanford University, 2008.

[67] J.M. Nyce and P. Kahn. *From Memex to Hypertext: Vannevar Bush and the Mind’s Machine*. Academic Press Professional, Inc., 1991.

[68] V. Otsason, A. Varshavsky, A. LaMarca, and E. De Lara. “Accurate GSM Indoor Localization”. In: *UbiComp 2005: Ubiquitous Computing* 3660 (2005), pages 141–158.

[69] J.R. Quinlan. *C4. 5: Programs for Machine Learning*. Morgan kaufmann, 1992.

[70] N. Ravi, N. Dandekar, P. Mysore, and M.L. Littman. “Activity Recognition from Accelerometer Data”. In: *Proceedings of the National Conference on Artificial Intelligence*. Volume 20. 3. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. 2005, page 1541.

[71] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava. “Using Mobile Phones to Determine Transportation Modes”. In: *ACM Transactions on Sensor Networks (TOSN)* 6.2 (2010), page 13.

[72] J. Rekimoto, A. Shionozaki, T. Sueyoshi, and T. Miyaki. “PlaceEngine: A WiFi Location Platform Based on Realworld Folksonomy”. In: *Internet conference*. 2006, pages 95–104.

[73] J. Rekimoto, T. Miyaki, and T. Ishizawa. “LifeTag: WiFi-Based Continuous Location Logging for Life Pattern Analysis”. In: *Proceedings of the 3rd international conference on Location-and context-awareness* (2007), pages 35–49.

[74] Tobias Rodehutskors. *Akustische Szenenerkennung auf Android-Mobilgeräten*. 2011.

[75] M.B. Russell, B.K. Rasmussen, J. Brennum, H.K. Iversen, R.A. Jensen, and J. Olesen. “Presentation of a New Instrument: The Diagnostic Headache Diary”. In: *Cephalalgia* 12.6 (1992), pages 369–374.

[76] R.R. Schaller. “Moore’s Law: Past, Present and Future”. In: *Spectrum, IEEE* 34.6 (1997), pages 52–59.

[77] R.E. Schapire and Y. Singer. “BoosTexter: A Boosting-Based System for Text Categorization”. In: *Machine learning* 39.2 (2000), pages 135–168.

[78] A.J. Sellen, A. Fogg, M. Aitken, S. Hodges, C. Rother, and K. Wood. “Do Life-Logging Technologies Support Memory for the Past? An Experimental Study Using SenseCam”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. 2007, pages 81–90.

[79] M. Shelkey and M. Wallace. “Katz Index of Independence in Activities of Daily Living (ADL)”. In: *The Gerontologist* 10.1 (1998), pages 20–30.

[80] T. Sohn, A. Varshavsky, A. LaMarca, M. Chen, T. Choudhury, I. Smith, S. Consolvo, J. Hightower, W. Griswold, and E. De Lara. “Mobility Detection using Everyday GSM Traces”. In: *UbiComp 2006: Ubiquitous Computing* (2006), pages 212–224.

[81] S. Sprager and D. Zazula. “A Cumulant-Based Method for Gait Identification using Accelerometer Data with Principal Component Analysis and Support Vector Machine”. In: *WSEAS Transactions on Signal Processing* 5.11 (2009), pages 369–378.

[82] L.M. Surhone, M.T. Tennoe, and S.F. Henssonow. *Gadex-Texte(Babylonien)*. VDM Publishing, 2010. ISBN: 9786133140615. URL: `http://books.google.de/books?id=KSrecQAACAAJ`.

[83] T.H. Tsien. *Paper and Printing*. Volume 5. Cambridge University Press, 1985.

[84] G. Tsoumakas and I. Katakis. “Multi-Label Classification: An Overview”. In: *International Journal of Data Warehousing and Mining* 3.3 (2007), pages 1–13.

[85] G. Tsoumakas, J. Vilcek, E. Spyromitros, and I. Vlahavas. “MULAN: A Java Library for Multi-Label Learning”. In: *Journal of Machine Learning Research* 1 (2010), pages 1–48.

[86] G. Tsoumakas, I. Katakis, and I. Vlahavas. “Random k-Labelsets for Multi-Label Classification”. In: *IEEE Transactions on Knowledge and Data Engineering* 23.7 (2011), pages 1079–1089.

[87] A. Varshavsky, M.Y. Chen, E. de Lara, J. Froehlich, D. Haehnel, J. Hightower, A. LaMarca, F. Potter, T. Sohn, K. Tang, et al. “Are GSM Phones THE Solution for Localization?” In: *7th IEEE Workshop on Mobile Computing Systems and Applications, 2006. WMCSA’06. Proceedings*. IEEE. 2005, pages 34–42.

[88] P.H. Veltink, H.B. Bussmann, W. de Vries, W.L. Martens, R.C. Van Lummel, et al. “Detection of Static and Dynamic Activities using Uniaxial Accelerometers”. In: *IEEE Transactions on Rehabilitation Engineering* 4.4 (1996), pages 375–385.

[89] S. Vemuri. "Personal Long-Term Memory Aids". PhD thesis. Massachusetts Institute of Technology, 2005.

[90] S. Vemuri, C. Schmandt, and W. Bender. "iRemember: A Personal, Long-Term Memory Prosthesis". In: *Proceedings of the 3rd ACM workshop on Continuous archival and retrieval of personal experiences.* ACM. 2006, pages 65–74.

[91] Y. Wang, J. Lin, M. Annavaram, Q.A. Jacobson, J. Hong, B. Krishnamachari, and N. Sadeh. "A Framework of Energy Efficient Mobile Sensing for Automatic User State Recognition". In: *Proceedings of the 7th international conference on Mobile systems, applications, and services.* ACM. 2009, pages 179–192.

[92] WebMD. *Webster's New World Medical Dictionary.* Webster's New World, 2008.

[93] M. Weller. *The Digital Scholar: How Technology is Transforming Scholarly Practice.* A&C Black, 2011.

[94] G. Wolf, A. Carmichael, and K. Kelly. "The Quantified Self". In: *TED – Ideas worth spreading: http://www.ted.com/talks/gary_wolf_the_quantified_self.html* (2010).

[95] Min-Ling Zhang and Zhi-Hua Zhou. "ML-KNN: A Lazy Learning Approach to Multi-Label Learning". In: *Pattern Recognition* 40.7 (2007), pages 2038–2048.

[96] M.L. Zhang and Z.H. Zhou. "Multi-Label Neural Networks with Applications to Functional Genomics and Text Categorization". In: *IEEE Transactions on Knowledge and Data Engineering* 18.10 (2006), pages 1338–1351.

[97] D. Zhao and M.B. Rosson. "How and Why People Twitter: The Role That Micro-Blogging Plays in Informal Communication at Work". In: *Proceedings of the ACM 2009 international conference on Supporting group work.* ACM. 2009, pages 243–252.

[98] Y. Zheng, L. Liu, L. Wang, and X. Xie. "Learning Transportation Mode from Raw GPS Data for Geographic Applications on the Web". In: *Proceedings of the 17th international conference on World Wide Web.* ACM. 2008, pages 247–256.

[99] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W.Y. Ma. "Understanding Mobility Based on GPS Data". In: *Proceedings of the 10th international conference on Ubiquitous computing.* ACM. 2008, pages 312–321.