Continuous and Energy-Efficient Transportation Behavior Monitoring

Samuli Hemminki

Helsinki October 1, 2012

Master's thesis

UNIVERSITY OF HELSINKI
Department of Computer Science

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty	Laitos — Institution — Department
Faculty of Science	Department of Computer Science

Tekijä — Författare — Author
Samuli Hemminki

Työn nimi — Arbetets titel — Title
Continuous and Energy-EfficientTransportation Behavior Monitoring

Oppiaine — Läroämne — Subject
Computer Science

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
Master's thesis	October 1, 2012	60 pages

Tiivistelmä — Referat — Abstract

In this thesis we present and evaluate a novel approach for energy-efficient and continuous transportation behavior monitoring for smartphones. Our work builds on a novel adaptive hierarchical sensor management scheme (HASMET), which decomposes the classification task into smaller subtasks. In comparison to previous work, our approach improves the task of transportation behavior monitoring on three aspects. First, by employing only the minimal set of necessary sensors for each subtask, we are able to significantly reduce power consumption of the detection task. Second, using the hierarchical decomposition, we are able to tailor features and classifiers for each subtask, improving the accuracy and robustness of the detection task. Third, we are able to extend the detectable motorised modalities to cover most common public transportation vehicles. All of these attributes are highly desirable for real-time transportation behavior monitoring and serve as important steps toward implementing the first truly practical transportation behavior monitoring on mobile phones.

In the course of the research, we have developed an Android application for sensor data collection and utilized it to collect over 200 hours of transportation data, along with 2.5 hours of energy consumption data of the sensors. We apply our method on the data to demonstrate that compared to current state-of-art, our method offers higher detection accuracy, provides more robust transportation behavior monitoring and achieves significant reduction in power consumption.

For evaluating results with respect to the continuous nature of the transportation behavior monitoring, we use event and frame-based metrics presented by Ward et al. [WLT06, WLG11].

ACM Computing Classification System (CCS):
C.2.4 [Computer-Communication Networks]: Distributed System,
H.3.4 [Information Storage and Retrieval]: Systems and Software,
H.4.m [Information Systems]: Applications-Miscellaneous,
I.5.2 [Pattern Recognition]: Design Methodology

Avainsanat — Nyckelord — Keywords
Mobile Sensing, Transportation Behavior Monitoring, Hierarchical Classification, Energy Efficiency

Säilytyspaikka — Förvaringsställe — Where deposited
Kumpula Science Library

Muita tietoja — övriga uppgifter — Additional information

Contents

1 Introduction

Modern smartphones are rapidly increasing in sensing capabilities and computational power. This combined with their ubiquitous presence, facilitated application development, and effective application distribution channels have enabled smartphones to mature into an attractive platform for human activity recognition [LML+10]. This thesis focuses on monitoring transportation behavior, a particular sub-area of human activity recognition. Our approach to transportation behavior monitoring is to design a mobile system, which automatically and continuously detects the user's active transportation modality in real time. We aim to provide detection on such fine granularity that all distinct motorised and pedestrian modalities are recognizable. The range of modalities depends on the transportation methods present at the target location. In our case, we implement the transportation behavior monitoring within Helsinki, a medium sized urban environment, in which case the relevant modalities are the different public transportation modalities (i.e., bus, train, tram, metro, car), pedestrian modalities (i.e., walking, running, biking) and stationarity.

Besides emerging as an interesting research topic on its own, benefits from efficient and robust transportation behavior monitoring would have impact on many research fields. For example, human mobility tracking would directly benefit from an ability to automatically monitor the transportation behavior of potentially large crowds of people [LAA+09, SQBB10]. Urban planning and in particular public transportation planning could gain from information on how the population is utilizing the different modalities of public transportation [ZLYX11]. Localization and positioning algorithms could be improved by constructing more elaborated motion models with the help of information of the user's current transportation modality [NBK10]. Persuasive applications could use the transportation behavior monitoring to automatically calculate, for example, CO_2-footprint or calorie consumption [FDK+09]. Finally, transportation monitoring could be used as part of user profiling, for examples, for real-time journey planning and guidance systems, or targeted advertising.

Smartphone-based transportation behavior monitoring is still a relatively new research area, and while current systems achieve high, over 90%, detection rates, these solutions still suffer from some fundamental limitations. First, the reported accuracies tend to ignore the continuous aspects of transportation behavior, typically using only simple frame-by-frame evaluation metrics (e.g., precision and recall). While these metrics give a good overview of the system's ability to estimate the right transportation modality at a given moment, they offer no insight into the systems

robustness or latency. Transportation activities tend to extend over relatively long periods of time and as such, metrics such as fragmentation rate and detection latency would add valuable perspectives to the evaluation [WLG11]. Second, current solutions provide only a coarse grained categorization for motorised transportation modalities, typically grouping all of them into one class. More fine grained classification of the motorised transportation modalities would provide information on public transportation behavior, applicable in many real-world applications. Finally, several of the other restraints in current solutions are related to the use of an integrated GPS receiver. The primary concern with GPS is its high power consumption, making use of the sensor unsustainable for long-term transportation behavior monitoring [LKLZ10]. Moreover, GPS works reliably only when it can establish unobstructed connection to the satellites, whereas many of the transportation modalities we would like to detect (e.g., metro, train and tram) prohibit reliable GPS connection. Furthermore, GPS suffers from an inconsistent delay between initiating the sensor and returning the first reliable value, inducing latency to the classification.

To address the limitations outlined above, this thesis provides the following contributions:

- We introduce a novel adaptive hierarchical sensor management scheme (HASMET) for increased control over sensor management and classifier control flow. We demonstrate that using this approach, we can i) substantially reduce the power consumption of transportation behavior monitoring and ii) improve classification accuracy and reduce classification fragmentation.
- We reduce the dependency on GPS by using other sensors and by incorporating more intelligent feature design. As part of the feature design, we introduce a novel method for estimating the gravity component from accelerometer measurements.
- We extend the range of motorised modalities that can be detected to cover common public transportation vehicles.

The rest of this thesis is structured as follows: First in Section 2 we give an overview of the existing research on related topics. In Section 3 we describe our data collection efforts, followed by a description of the sensors and features considered in our work. Next, in Section 4 we introduce the architecture of the hierarchical classifier, the strategy for energy-efficient sensor management and our methodology for sensor and feature selection. In Section 5 we introduce our evaluation metrics, followed by the

results. Finally, we conclude the thesis in Section 6 with a summary of our work and ideas for future development.

2 Related work

Extensive research exists on many of the areas related to our work. Our proposed system combines efforts from studies on human activity recognition, particularly from the subfield of transportation behavior monitoring, and methods of energy-efficient sensor management. Below we give a brief summary of each of these areas and discuss how this thesis extends the previous research.

2.1 Activity Recognition

Research on human activity recognition is a long-standing, widely studied field within the wearable and ubiquitous sensing domain. The existing research can be divided into three categories based on the technological approach [CMT+08]. The oldest and most common approach is the use of *research prototypes*, tailored sensor systems designed for particular study or series of them. At simplest, a research prototype can consist of one or more attachable sensors worn at specific locations on the user. The accuracy yielded with this approach, especially with a multiple sensor strategy, have generally been high [KSS03, LWJ+04, BI04]. However, until the sensing units can be embedded on a user in a non-obtrusive way (e.g., in user clothing), such strict and cumbersome sensor placements render these systems impractical for everyday use. A more suitable domain for these systems are the specific situations where high accuracies and fine-grained activity classes are required. An example of such a situation is presented by Jakob et al., who use a set of attached accelerometer sensors to recognize different phases in surgical procedures [BDJN11]. An alternative approach to activity recognition is to combine several sensing units into one platform. One of the most versatile such systems is the Mobile Sensing Platform [CBC+08], with sensing capabilities for three-dimensional acceleration, air pressure, humidity, visible and infrared light, temperature, audio and orientation. The platform was used, among other studies, in a work evaluating the effect of awareness of the user's daily physical activity [CKM+08]. While less cumbersome than multiple attached sensors, these approaches require external, often expensive hardware, making them infeasible to apply on large scale.

A step towards wider applicability is to implement the activity recognition using one of the existing *commercial devices*, specifically designed for this task [CAH08]. The most simple and widespread of such devices are the pedometers [CMT+08], instruments for calculating steps by detecting gait from the user's walking motion.

Pedometers can be applied to human activity recognition in several methods, most commonly: i) to measure user's physical activity from daily step count, ii) to identify persons based on their physical activity or iii) to identify temporal distribution of daily activities [SCB04]. As an example, Tudor-Locke et al. use pedometers to calculate daily step counts as an indicator for a healthy lifestyle [TLBJ04]. In a related work, Chan et al. investigate health benefits of persuasive pedometer-based physical activity monitoring [CRTL04]. More complex commercial devices can contain multi-axis accelerometers (e.g., FitBit[1], Tracmor[2]) and multiple sensors (ImpactSport[3]) capable of sensing more elaborate properties such as heart rate, heat flux, galvanic skin response and skin temperature. In a work combining several commercial sensing devices, Könönen et al. use a Suunto wrist-top computer, a Garmin GPS device, an Embla external audio recorder and an iPaq PDA to collect a rich feature set and use it to recognize between nine different physical activities [KMS+08]. These systems, while using available off-the-shelf devices, lack a standard platform and effective distribution channels; thus they are not consumer friendly enough to achieve a large user base.

The most recent shift in the wearable activity recognition communities has been towards using smartphones as the primary platform. There are several qualities that make smartphones attractive. From a practical point-of-view, smartphones are widespread and carried continuously by the users, thus providing for a potentially non-intrusive method for continuous and large-scale activity recognition [LXL+11]. From a technical point-of-view, the extensive and rapidly increasing computational and sensing capabilities of modern smartphones enable accurate activity recognition [LML+10]. From the developer's point-of-view, application development and distribution has been significantly facilitated with improved programmability and effective distribution channels [LML+10]. As an example of smartphone-based activity recognition, Lau et al. conducted research on patient monitoring [LKD+10]. In their work, a mobile phone embedded accelerometer was employed to detect coarse-level patient activities, such as 'sleeping', 'awake' or 'moving and doing exercises'. In another example, Papliatseyeu and Mayora investigated human activity recognition based on a fusion of smartphone sensors (GPS, GSM, Wi-Fi and Bluetooth) [PMI08]. A collection of different sensors was used as opposed to a single sensor to compensate for sensor-specific limitations.

[1] `http://www.fitbit.com/` [Retrieved: 2012-08-06]
[2] `http://www.tracmor.com/` [Retrieved: 2012-08-06]
[3] `http://impactsport.org/` [Retrieved: 2012-08-06]

2.2 Transportation Behavior Monitoring

Transportation behavior monitoring can be considered a special subfield in the wider domain of activity recognition. The task of detecting the current transportation modality itself can be further divided into two subtasks: i) stationary detection, i.e., detecting the user as stationary or mobile, and, when the user is detected as mobile, ii) locomotion recognition, i.e., recognizing the user's locomotion method. Stationary detection is a well studied concept, which has been approached with varying methods. For example Krumm and Horvitz [KH04] utilized characteristics of the received signal strength indicator (RSSI) from Wi-Fi access points to classify the user as mobile or stationary. Wi-Fi based stationarity detection was further explored by utilizing the frequency domain of RSSI values by Muthukrishnan et al. [MLMH07]. In a more recent work, accelerometer-based stationary detection was added to the Wi-Fi detection for energy-efficient sensor management by Kim et al. [KKES10]. This approach was further extended with an addition of a speed-threshold to include motorised vehicles by Kjaergaard et al. [KBBN11].

The latter task, locomotion recognition, has been studied in varying degree in the domain of activity recognition. Typical locomotion types include different pedestrian modalities (e.g., walking, running or moving in stairs), non-motorised assisted transportation modalities (e.g., bicycling, roller skating) and motorised transportation modalities (e.g., car, bus, train). In one of the earliest works, Farringdon et al. created a system based on attachable accelerometers to identify stationary, walking and running activities [FMT+99]. An extensive activity recognition study, including a variety of locomotion activities, was conducted by Bao et al. [BI04]. In their approach, five biaxial accelerometer were attached at specific locations of the user to recognize between 20 activities, reaching an average accuracy of 84%. Locomotion detection with a single triaxial accelerometer was investigated by Ravi et al. [RDML05], focusing on comparing the efficiency of different classification techniques. The accuracies reported by this study ranged from 57% with a decision table, to up to 90% with plurality voting for recognizing between eight activities.

While the detection accuracy of systems based on wearable accelerometer(s) are relatively high, they are robust only for modalities involving vivid kinematic motion. Activities with low intensity kinematic motion, such as stationary and motorised modalities, are significantly harder to separate with solely accelerometer based solutions [RMB+10]. An effective approach for distinguishing between motorised transportation modalities and stationarity is to monitor changes in the user's

environment and location. For example, changes in received signal strengths from GSM radio were used by Sohn et al. [SVL+06] to discern coarse-grained transportation modalities (stationary, walking, driving). The GSM features were augmented with RSSI values from Wi-Fi access points to detect more subtle changes in location by Mun et al. [MEBH08]. These approaches yielded accuracies of 80−90% for motorised transportation detection. However, they rely on features that have been extracted over data windows with a duration from 40 seconds to 2 minutes, which makes these approaches unsuitable for applications that require (near) real-time information about transportation behavior. An example of such an application is a high-accuracy positioning system, which uses the user's current transportation modality to estimate the user's speed and trajectory. Another, more direct method for detecting changes in user's location has emerged with the expanding availability of GPS sensors in modern mobile phones. This option was explored by Zheng et al. [ZLC+08], who detected transportation modalities based on features extracted solely from GPS. In addition to speed and location information, Zheng et al. extracted three novel features from GPS: heading change rate, stopping rate and velocity change rate. A more detailed presentation on these features is given in Section 3.3, page 22. Using this approach together with spatial information Zheng et al. reached average accuracy of 76% in classifying between stationarity, walking, biking, driving and traveling by bus. Using the GPS is an attractive option due to its easy access and accurate, near real-time updates. The GPS, however, also has some drawbacks which limit its usability: it has a high energy profile, requires an inconsistent time period to achieve a satellite lock and is unusable or unreliable when a clear view to the satellites is obstructed. Furthermore, several of the benefits of GPS require external GIS-information, such as route information of a public transportation vehicle, which might not always be readily available.

The current state-of-the-art solutions for transportation behavior monitoring use multiple sensors for the detection task. We consider one of such systems, proposed by Reddy et al. [RMB+10] as our primary baseline to compare our solution against. The system utilizes a combination of GPS and accelerometer sensors to classify the user's transportation modality as: stationary, walking, running, biking or traveling with a motorised vehicle. Besides high accuracy, we choose this work as our primary baseline due to its emphasis on user convenience. Some of the novel qualities presented in the approach of Reddy et al. include: unconstrained phone orientation and position, cross-user usability and independence from external information. As a secondary baseline, we compare our solution against the system proposed by Mun

et al. [MEBH08]. We chose to add this as a second baseline since it utilizes similar GSM and Wi-Fi features as our solution. In our work, we continue along the direction of Reddy et al. emphasizing real world applicability by reducing the energy consumption of the detection task, while simultaneously offer improved detection accuracy. Our work additionally provides finer granularity of detectable motorised transportation modalities and improves the overall consistency of the transportation behavior monitoring.

2.3 Energy Efficiency

While the sensing and computational capabilities of smartphones have been rapidly increasing, the battery capacity has developed at a much more conservative rate, limiting the use of the phone's new capabilities. As such, energy efficiency has emerged as a vital property for continuous smartphone-based sensing systems. The research on energy efficiency has tackled this problem from two directions: by reducing general power consumption at the hardware and system level, and by proposing energy-efficient application designs.

Examples of approaches that address energy-efficiency at the hardware level include the delayed data transmission strategy of Kravets and Krishnan [KK98] and the diminished data quality approach of Flinn and Satyanarayanan [FS99a,FS04]. Both techniques focus on reducing power consumption by manipulating the data transmission between communication components. The former technique, delayed data transmission, improves energy-efficiency by sending data less frequently in larger ensembles as opposed to frequent, small data packages. The study reports reduction in power consumption of up to 83% for communication components. The latter technique, diminished data quality, adapts the data quality in accordance with the remaining battery charge. While specific energy saving depends on the extent of data quality diminishing, Flinn and Satyanarayanan report battery recharge interval increase of up to 30% without sacrificing application quality. Another system-level scheme based on the remaining battery charge is the battery-driven power management technique proposed by Benini et al. [BCMS01]. The system tailors a group of dynamic power management policies with the current battery charge, turning off resources during periods of inactivity. Using this technique, Benini et al. report up to a 66% extension in battery recharge intervals. Two further related approaches are the dynamic voltage scaling (DVS) and dynamic frequency scaling (DFS), also known as CPU throttling [PS01]. These techniques, often used in conjunction, re-

duce the power consumption of the system by adjusting the computational power, i.e., processor frequency and system voltage, dynamically to match the current system load.

Research on application level energy efficiency has already seen wide attention within the mobile sensing domain [RSHI08,KBBN11,SC10,MPF+10]. Most notable reduction in power consumption has been achieved by utilizing low-power sensors in place of more consuming ones (e.g., by replacing GPS with accelerometer and GSM). Further energy reductions are attainable by employing duty cycling on the sensors, i.e., by decreasing the sampling frequency of the sensors when applicable. In a work closely related to ours, Wang et al. propose a hierarchical sensor management strategy, coined as the Energy Efficient Mobile Sensing System (EEMSS) [WLA+09]. The EEMSS employs a minimal set of sensors to detect the user's state and then tracks the sensors for a trigger condition to initiate state transition. The system is evaluated by detecting three states, 'At some place', 'Walking' and 'Vehicle', achieving accuracies of 99%, 84% and 74% respectively. Compared to systems which periodically sample the sensors, EEMSS can extend the lifetime of the device by approximately 75%. The latency of detecting a state transition, however, ranges from 40 seconds to 5 minutes, making the system unsuitable for applications that require near real-time information. Another example is EnTracked, an energy-efficient position tracking system proposed by Kjaergaard et al. [KLGT09]. The main novelty of the work is to reduce the use of GPS sensors by specifying an error bound, adjustable by applications, within which the position accuracy is required. The trigger condition for GPS sampling, i.e., the instant when the error-limit is exceeded, is calculated from speed estimation and accelerometer-based motion analysis.

Within the field of transportation behavior monitoring, only limited amount of work on energy efficiency has been conducted. Some suggestions and initial work however exist. In addition to work referred above, Reddy et al. estimated the energy footprint of their system by measuring power consumption of the employed sensors and the classifier over five 20-minute-trials of each the of activity classes [RMB+10]. They further implemented a simple method for improving the classifiers' energy efficiency by detecting switches from indoor to outdoor settings, and turning off the classifier while indoors. A system based on predicting the next likely set of activities of the user was introduced by Gordon et al. [GCMB12]. Based on the predicted activity set, only the relevant sensors are operated. The method achieves substantial (reporting up to 84%) energy savings while maintaining comparable accuracy. Additionally, there exists several low-energy systems capable of limited transportation behavior

monitoring based solely on accelerometer [WCM10, LYL+10]. A method to further decrease the energy consumption of a accelerometer-based system was proposed by Yan et al. [YSC+12]. The system, coined the Adaptive Accelerometer-based Activity Recognition (A3R), works by dynamically adjusting the frequency rate of the accelerometer depending on the prevalent activity, achieving energy savings of approximately $20 - 25\%$. In our research, we improve the energy-efficiency of the monitoring task by constructing an energy model for each of the sensors and finding the most energy-efficient sensor combination for each of the hierarchical classification phases. Finally, we design a sensor management system to remove unnecessary sensor polling.

3 Data collection

Over the course of studying and developing the transportation behavior monitoring, we have collected over 200 hours of data from various transportation modalities. The data has been collected with two smartphone models: Samsung Nexus S and Samsung Galaxy S II. For evaluating the energy footprint of our approach, we have also collected measurements about the energy requirements of different sensors on our target smartphones. In the following we describe the collected datasets in detail.

3.1 Transportation Behavior Data

For the transportation behavior data collection task we have developed an Android application, SensorLogger, capable of monitoring and storing data from the available sensors on the smartphones: Accelerometer, Bluetooth, GPS, GSM, Gravity sensor, Gyroscope, Light sensor, Linear acceleration sensor, Magnetometer, Microphone, Orientation, Proximity sensor, Rotation vector and Wi-Fi (see Section 3.3 for further discussion of the sensors). The application was installed on two models of smartphones: Samsung Galaxy S II and Samsung Nexus S. The two models were attached to form a single sensing unit; see Figure 1.

Figure 1: Sensing unit consisting of Samsung Galaxy S II and Samsung Nexus S.

We opted for using a combination of two phones due to model-specific limitations. At the time of our research, the Galaxy S II phone was unable to provide detailed GSM information, while the Nexus S lacked many of the sensors present in the later

models. Additionally, at the time of development, the Wi-Fi sensor on the Galaxy S II model always scanned both 2.4 GHz and 5 GHz frequency ranges. Since in our study we focus only on the 2.4 GHz range, this induces unnecessary time and energy cost to the sensor's operation. As a result, we used the Nexus S phone to scan the GSM and Wi-Fi information, while the Galaxy S II phone was used to scan the rest of the available sensors. The raw sensor data was annotated with ground truth labeling of the current transportation modality. The ground truth annotation was performed manually with a help of a specifically designed Android application. In order to not disturb any of the sensing units, an additional phone was used for assigning the ground truth labels.

The data collection was carried out with two methods:

- Collecting long data traces from everyday transportation behavior. This dataset was collected by two individuals over period of three months, with a total of over 170 hours of collected data.
- Collecting data while following a predefined scenario, detailed below. The scenario data was collected by six different users over period of two weeks, with a total of 29 hours of collected data.

While the bulk of the data was collected by individual users from everyday transportation, this data was mainly used for developing, validating and experimenting with transportation behavior monitoring. The final evaluation was conducted with the data collected from the predefined scenario. Since our scenario does not include sections of driving, biking or running, the scenario dataset was supplemented with 5 hours of data from the everyday data collections containing measurements from these modalities. To obtain and ensure phone location independence, and according to prevailing best practices, the data was collected simultaneously from several mobile phone placements. The placements considered were the three most common locations for a mobile phone in an urban environment [ICG05]: trouser pocket, bag, and jacket pocket. As a result, each data collection task involved seven phones: three sensing units consisting of two phones each used for collecting the measurements and one phone used for assigning the ground truth labels.

Scenario The scenario (see Figure 2 for an overview) took place in Helsinki, Finland. Each data collection case involved two persons: the participant and the supervisor. The participant was equipped with the sensing units, while the supervisor

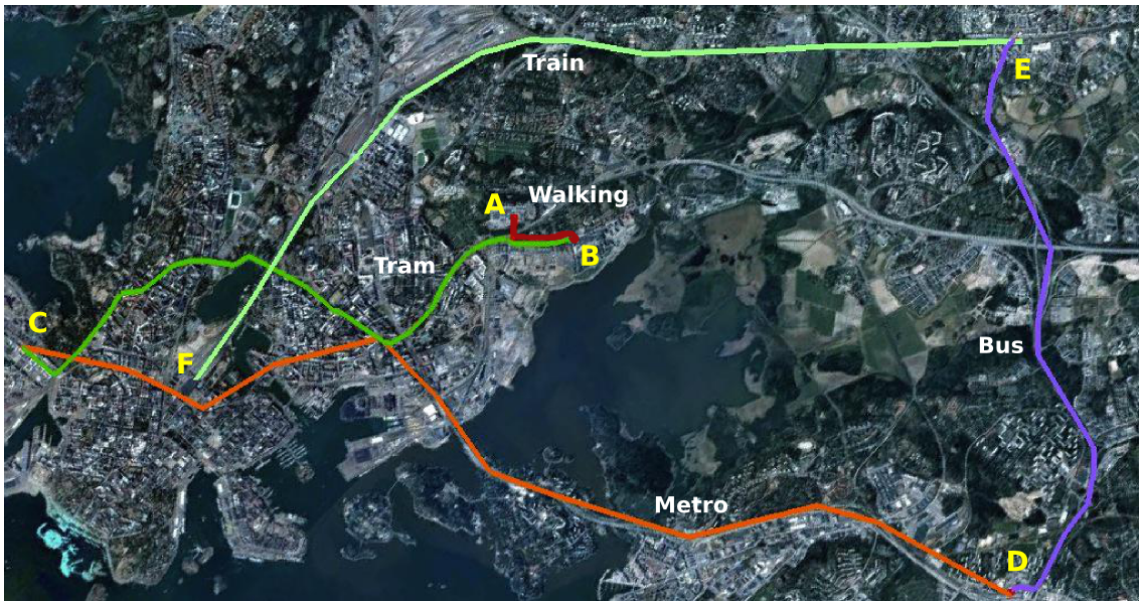

Figure 2: Overview of the scenario used in our data collection.

was responsible for inserting the ground truth labels and guiding the participant through the scenario. The public transportation modalities available in Helsinki are bus, train, tram and metro. The scenario was planned to contain a span of 20 minutes from each of these available transportation modalities. In addition, a walking section of similar length was added to the beginning of the scenario, resulting in a total length of approximately 120 minutes. The initial walking section (A $\rightarrow$ B in Figure 2) contained a small downhill, short section of stairs and two traffic lights. The tram (B $\rightarrow$ C) passed though a busy, often congested section of the downtown. After the tram, a section of long escalators were traveled to gain access to the metro (C $\rightarrow$ D). The bus section (D $\rightarrow$ E) was a combination of slow-moving downtown travel, followed by a short period of motorway travel. Finally, the train (E $\rightarrow$ F) section used the main railroad line, finishing the scenario at the central railway station. Between each motorised transportation modality, there was a brief walking segment and typically 3-10 minutes of stationarity while waiting for the vehicle of the next transportation modality. After the data collection, the data was revalidated to correct any clearly erroneous ground truth labels resulting from human error.

The data collection was carried out during the winter of 2011-2012 in harsh conditions, which limited the feasibility of collecting sufficient amounts of running and biking data. Hence, these modalities are only used for training the classifier and excluded from the testing set. However, previous work [BI04, RMB+10], as well as small scale experiments that we have conducted with the supplementary data

TRAINING		
Method	**Persons**	**Duration**
Scenario	4	17h 25min
Individual	2	2h 50 min
TOTAL	**5**	**20h 15 min**

TESTING		
Method	**Persons**	**Duration**
Scenario	2	11h 15min
Individual	2	1h 50 min
TOTAL	**4**	**13h 05 min**

Table 1: Summary of the datasets used in the evaluation.

indicate that these modalities can be easily identified with high accuracy.

For evaluation purposes, we split the data into separate training and testing sets, summarized in Table 1. The training set contains data collected from four participants, and 60% of the supplementary data. The remaining data is used for constructing the test set. This division of distinct users for train and test sets, and various sensor placements also functions as an indicator of the generalization performance of our system across various users and sensor placements.

3.2 Energy Consumption Data

Sensor energy consumption data was collected using the Monsoon Power Monitor[4], a device capable of accurately measuring the power of any device using a single lithium battery; see Figure 3 for the use of the device to measure energy consumption of Samsung Galaxy S II. All energy measurements were collected by supplying the phone with a constant voltage of 3.9 V and tracking the current (mA) with frequency of 5000 Hz. To minimize noise and disturbances in the collection of energy measurements, the screen of the device was turned off and all background applica-

[4] `http://www.msoon.com/LabEquipment/PowerMonitor/` [Retrieved: 2012-09-04]

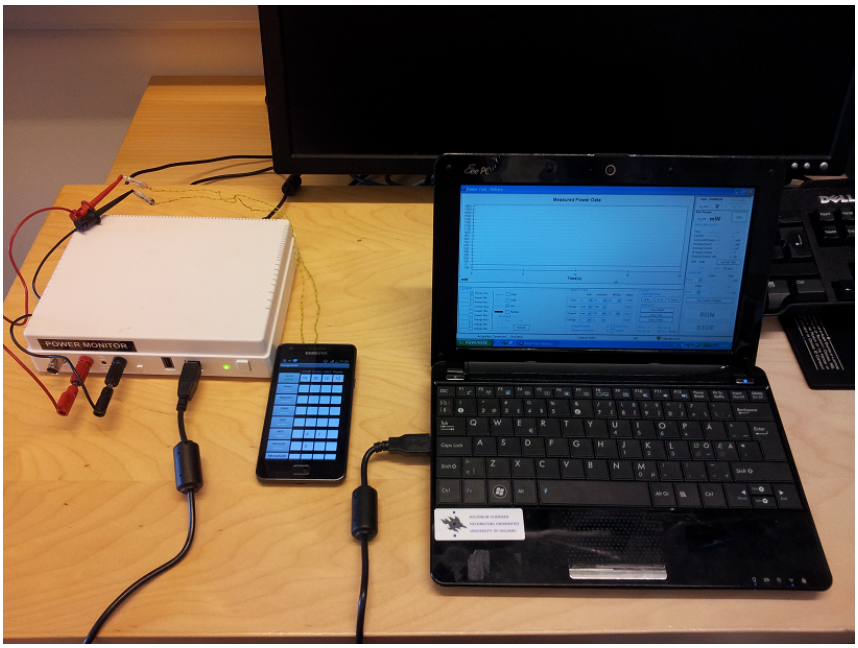

Figure 3: Our setup with the Monsoon Power Monitor with a Galaxy S II phone.

tions were closed. Additionally, to further reduce the noise caused by the phone's background operations, the phone was set to airplane mode for all the sensors excluding Wi-Fi and GSM, for which normal mode was required. For each sensor, we programmatically switch the sensor on, execute a series of 10 sampling sequences with idle periods in between and finally turn the sensor off. The durations of the sampling sequences and the idle phases varied according to the complexity of the sensor's energy profile, ranging between $30-60$ seconds and $10-30$ seconds respectively. This procedure was repeated twice for each sensor, resulting in approximately 2.5 hours of energy measurements.

Power Models From the energy measurements, we have constructed simple empirical power models for the purposes of energy consumption evaluation. Specifically, for each sensor, we have identified the distinct stages corresponding to switching the sensor on/off, sampling the sensor, and maintaining the sensor in idle mode (i.e., maintaining the sensor on but not actively sampling). Sensors with a complex energy profile are further divided into more fine grained sampling stages. For example, in the GPS sensor's energy profile we observe two separate pre-sampling stages (see Figure 4), which correspond to initiating the sensor and the phase when GPS is achieving a satellite lock.

While the power consumption for sampling is approximately constant for most sen-

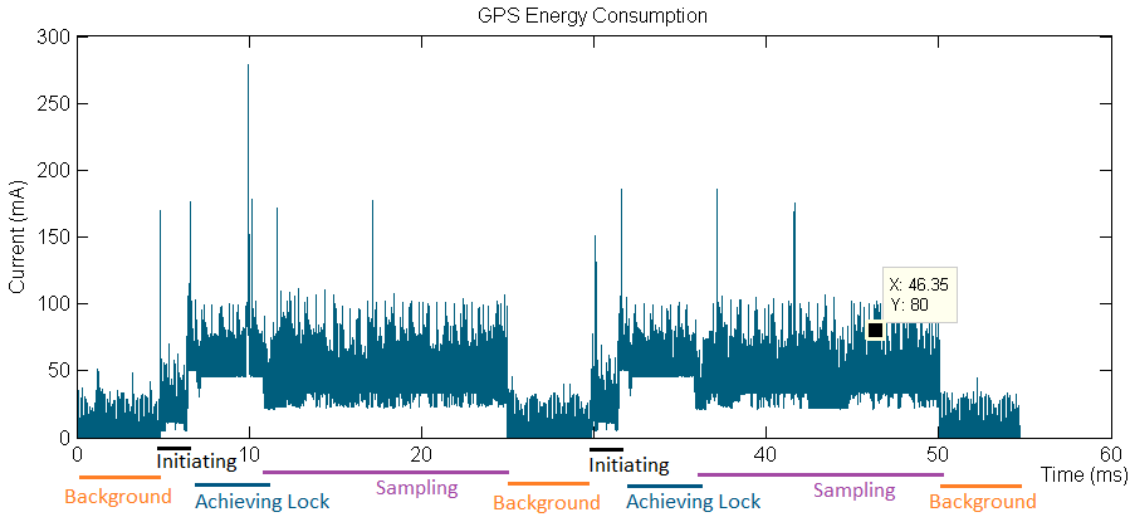

Figure 4: Current (mA) associated with sampling the GPS sensor at 3.9 V, repeated twice for 20 seconds.

sors, the wireless communication sensors, Wi-Fi and Bluetooth, have varying consumption. The consumption depends on the number of visible access points (Wi-Fi) or discoverable devices (Bluetooth). For Wi-Fi, however, this is reflected in the duration of the scan and thus it suffices to observe and model the duration of the scan. For Bluetooth, the energy consumption increases roughly linearly with the number of discovered devices and a simple linear function can be formulated between the number of devices and the energy cost:

$$Bluetooth_{energy} = 797 + 55\text{*number of discovered devices.}$$

Note that for GSM, proximity and light sensors, the power consumption resulting from sampling is effectively zero as the phone already constantly polls this information.

After identifying the various sensor stages, we estimate the power consumption (mW) for each stage. As we are primarily interested in estimating the overall energy consumption, simple mean value of the measured current is used to represent each stage. In case the sensor stage has a fixed duration, the total energy (mJ) of the stage is additionally calculated. For derived values corresponding to each sensor, see Table 2.

Sensor	Switch ON	Switch OFF	Sampling	Idle	Pre-Sampling
Accelerometer	-	-	21.1mW	-	-
Gravity	-	-	25.31mW	-	-
L.Acceleration	-	-	25.86mW	-	-
Magnetometer	-	-	53.35mW	20.9mW	-
Orientation	-	-	52.73mW	20.4mW	-
Rotation	-	-	55.26mW	21.5mW	-
Gyroscope	-	-	154.4mW	22.6mW	43.9mJ
Microphone	123mJ	35.8mJ	101mW	-	-
GPS	77.2mJ	-	115mW	-	168mW
Bluetooth	480mJ	381mJ	797+55mJ*	3.17mW	-
WLAN	1292mJ	615mJ	293mW	12.3mW	-

*: Energy increase per device.

Table 2: Energy consumption of different stages of the sensors

3.3 Sensors and Features

As a part of our research, we have investigated several sensors to identify the ones most efficient for the task of transportation behavior monitoring. In the following we briefly describe the sensors we considered and the features we extracted from the sensor values. As our target is to support applications that require near real-time information of transportation behavior, we are extracting the features from windows of short (1.2 seconds) time span. Below we give a listing of all the sensors and features considered. A more detailed examination of the selected features is given in Section 4.2.1.

Accelerometer The accelerometer chip in our use is the Galaxy S II embedded ST Microelectronics LIS3DH, capable of providing three-dimensional acceleration measurements (reported in m/s^2). We sample the accelerometer at 100 Hz frequency, which corresponds to the maximum of the phone's capability.

As the first preprocessing step, we envelope the measurements into 1.2 second windows with 50% overlap. Next, to minimize high frequency noise from the data we apply a low-pass filter on the measurements along each of the axes. The filter is an energy-threshold filter, which eliminates the highest frequencies corresponding to 10% of the window's total energy. After filtering, we construct three repre-

sentations from the accelerometer values: the total magnitude of accelerometer, the horizontal magnitude and the vertical magnitude. Magnitude is extracted using the L_2 norm, defined as $a = (a_x^2 + a_y^2 + a_z^2)^{1/2}$. The L_2 norm is widely used in activity recognition to minimize the effects of orientation to accelerometer features [RMB+10, WCM10, LYL+10]. The horizontal and vertical components are derived from the data by i) determining the accelerometer's gravity component g and ii) by utilizing g, computing projections along horizontal and vertical planes.

For estimating the gravity component, we use an extension of the technique proposed by Mizell [Miz05]. In Mizell's approach, gravity is estimated with the mean of accelerometer measurements over a duration of a few seconds. This approach, however, lacks robustness when sustained, directional acceleration is applied on the accelerometer, or when the accelerometer orientation changes rapidly. As both of these situations are fairly common in transportation behavior, we have extended the technique with two methods. To address the problem with sustained acceleration, we use the variance of the accelerometer measurements to detect near stationary periods, and estimate the gravity component opportunistically when the variance is low. A precision score p denoting the amount of variance is attached to the gravity estimate, forming a tuple $\{G_{est}, p\}$. The gravity estimate is updated only when the variance of the current window is lower than the value of p. To ensure that our gravity estimate remains up to date, the p value of the prevalent gravity estimate is increased with a small constant (currently we use 0.00001) each time the current data window's variance is higher than the value of p. To reduce erroneous estimates resulting from changes in phone orientation, we reset the tuple $\{G_{est}, p\}$ when a large shift in orientation is observed. Shifts in orientation are detected by comparing the prevalent gravity estimate against the mean of the current measurement window. Whenever these differ by more than a predefined threshold (currently we use 2 m/s) along any of the axes, we assume the orientation of the device has changed and re-initialize the gravity estimate for each axis to zero; see Figure 5 for a comparison between our gravity estimate and that of Mizell's approach. From the figure, we can observe the two cases mentioned above: in case of sustained acceleration (span A) the gravity estimation of Mizell's approach follows the measured acceleration too closely. In case of orientation changes (span B), the system has a transition phase, within which the gravity estimates give highly erroneous estimates.

Once the gravity component $\vec{g} = (g_x, g_y, g_z)$ has been estimated, we derive the vertical projection $\vec{v}$ and the horizontal plane projection $\vec{h}$ from the observed acceleration $\vec{a} = (a_x, a_y, a_z)$. First the dynamic acceleration $\vec{d} = (a_x\text{-}g_x, a_y\text{-}g_y, a_x\text{-}g_x)$ is computed,

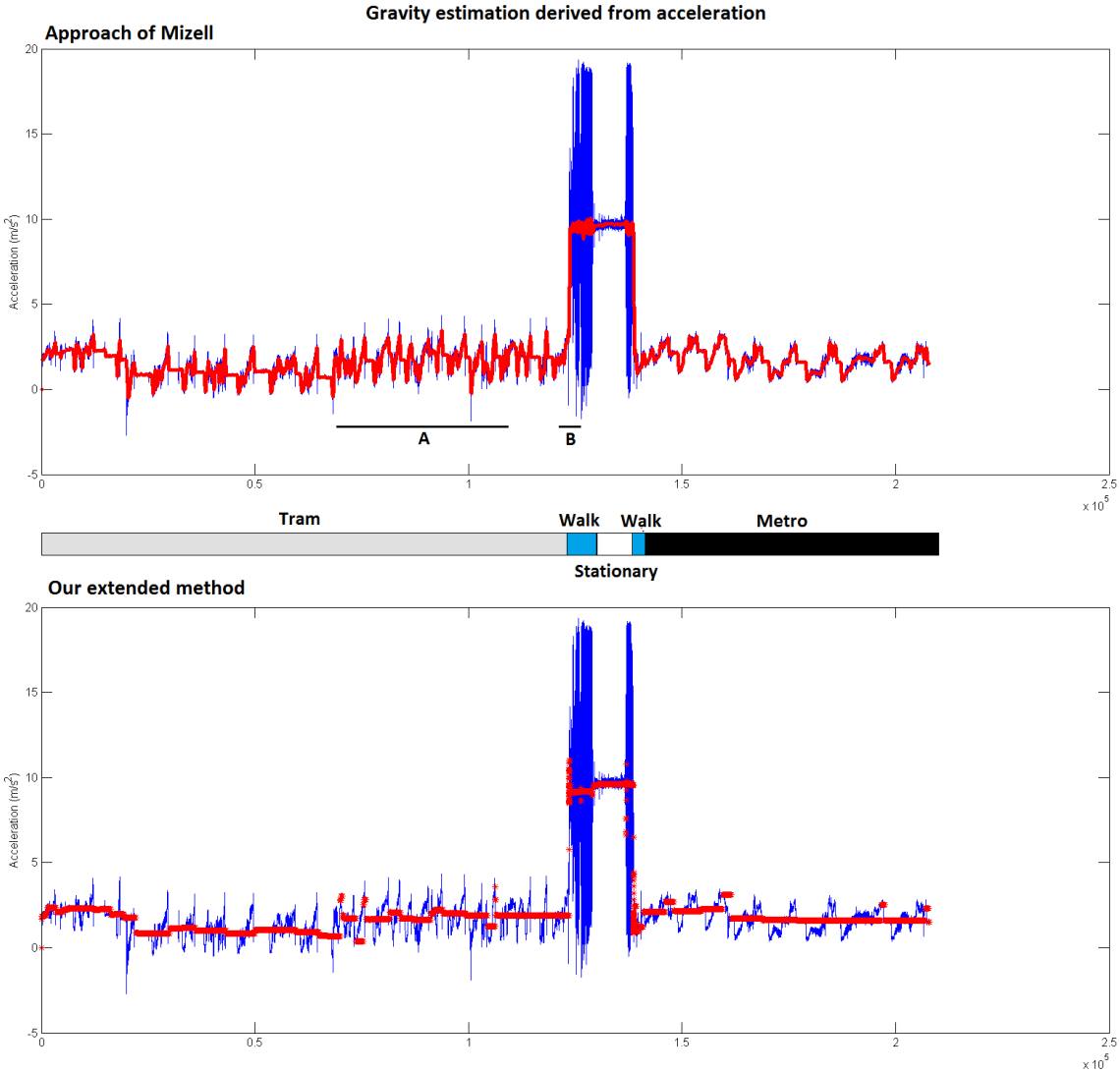

Figure 5: Comparison of a single accelerometer axis (blue) with the estimated gravity component (red) using Mizell's approach (upper picture) versus using our extended method (lower picture). From the figure, we can observe that in case of sustained acceleration (span A), the estimated gravity component from Mizell's approach tracks the accelerometer values too closely. Additionally, in case the orientation changes (span B), the approach of Mizell produces greater errors before reaching a reliable estimate.

Domain	Features
Statistical	Mean, Variation, Standard Dev., Min, Max, Median, Range, IQR, RMS, Kurtosis, Skewness, Autocorrelation, Mean Pairwise Correlation, Mean Cross-Correlation
Time	Integral, Double-Integral, Zero-Crossings
Frequency	FFT: DC, 1Hz, 2Hz, 3Hz, 4Hz, 5Hz, 6Hz, Energy, Entropy, Maximal Coefficient, Wavelet Entropy (Haar2), Wavelet Magnitude

Table 3: Accelerometer features by domain. For more detailed information of the different features, see the extensive feature study by Figo et al. [FDFC10].

which is the gravity-eliminated acceleration caused by the user's movement. Now the vertical component $\vec{v}$ is the projection of $\vec{d}$ upon the gravity component $\vec{g} : \vec{d} = \frac{\vec{d}\cdot\vec{g}}{\vec{g}\cdot\vec{g}} * \vec{g}$. The horizontal component is simply the subtraction of the dynamic acceleration and the vertical component $\vec{h} = \vec{d} - \vec{v}$. Finally, to obtain orientation independence, we calculate the L_2 norm for both $\vec{v}$ and $\vec{h}$.

After the preprocessing steps described above, we extract an extensive set of features from each of the three representations introduced above; for a full list of features, see Table 3. Our feature space is based on work by Figo et al. [FDFC10] and encompasses statistical metrics (e.g., mean, variance, kurtosis), time-domain metrics (e.g., double integral, cross-correlation and zero crossings) and frequency-domain metrics (e.g., energy, first six FFT components, entropy, wavelet decomposition and the sum of FFT coefficients). The feature space dimensionality is 29 features per accelerometer representation, i.e., the total number of accelerometer features extracted is 87.

GSM The Android API available on our smartphones provides an interface for monitoring changes in the GSM signal environment. The information that can be accessed is limited to the received signal strength and identifier of the cell tower that the phone is currently connected to. At the time of development, no method for polling the GSM information at a defined frequency was available. Instead, new values are updated every time a change in the signal strength is detected. In practice, this results in a varying sampling rate between 0.3 and 0.5 Hz.

For feature extraction, we aggregate measurements using a five second sliding win-

dow with 50% overlap. From each window we extract nine GSM features based on the works of Sohn et al. [SVL+06] and Mun et al. [MEBH08]: The mean and variance of the received signal strength (RSS) values from each observed cell tower, the handover count, dwell-time (i.e., time elapsed since the last handover), the number of unique cells present within the window and the ratio of common cell towers between consecutive windows. We also compute the KL-divergence between the ratio of common access points between consecutive windows, which Nurmi et al. [NBK10] have shown to be a good indicator of radio map divergence in GSM positioning. The KL-divergence between probability distributions p and q is defined as $D_{KL}(p \parallel q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$, where p and q can be any arbitrary probability distributions. Finally, we construct a time-interpolated handover score $GSM_{handover}$, which increases at each detected handover and decreases over time when no handovers are observed. Specifically, $GSM_{handover}$ is calculated as follows:

- $GSM_{handover}$ initializes from a value of one.
- Each time a handover is detected, the score is increased: $GSM_{handover} = 5 + GSM_{handover} * 0.5$.
- If no handover is detected within a specific duration (currently 15 seconds) the handover score is decreased: $GSM_{handover} = GSM_{handover} * 0.5$.

This approach results in a function, which increases asymptotically towards a value of 10 when GSM handovers are detected frequently, and decreases asymptotically towards a value of zero when no handovers are detected. The constants for the formula were selected empirically to provide optimal separation between stationary, pedestrian and motorised transportation modalities.

Wi-Fi To monitor changes in the Wi-Fi signal environment, the 2.4 GHz frequency range was scanned periodically using the Nexus S smartphone. The Nexus S phone was used for this task instead of the Galaxy S II due to the latter always scanning additionally the 5 GHz frequency range[5], which induce longer scanning times and higher energy consumption. Each new scan was triggered one second after the previous scan was completed. As the duration of a single scan is not constant, the actual sample rate varies between 0.4 Hz and 1 Hz. From each scan, we extract the

[5]Note that later versions of the software have enabled an option to select the preferred bandwidth.

MAC address, signal strength, exact frequency (i.e., Wi-Fi channel) and the SSID from each visible access point. For evaluating the energy cost of a single scan, we also record the duration of the scan. As a preprocessing step, we we detect and tag universally administered wireless access points [KKES10] to prevent on-board wireless access points, installed for example on trams and subways, from interfering with the Wi-Fi features.

The feature extraction follows closely the procedure with GSM. First, we aggregate the information of Wi-Fi scans over a five second window. Next, we calculate the mean and variance of the RSSI-values for each visible access point. Additionally, we detect the dominant access point within the window (i.e., the AP with highest mean signal strength), calculate its dwell-time [MEBH08], signal strength variance [KH04], and the ratio between the strongest and the weakest access point's signal strengths [KM08]. Finally, as with GSM, we calculate an interpolated dwell-time score $WLAN_{dwelltime}$, which increases when the dwell-time of the dominant access point increases and decreases when the dwell-time of the dominant access point decreases (i.e., when the access point is no longer visible). Specifically, the $WLAN_{dwelltime}$ is calculated as follows:

- $WLAN_{dwell-time}$ initializes from a value of one.
- Each time the dwell-time of the dominant access point increases, the score is increased: $WLAN_{dwelltime} = 1 + WLAN_{dwelltime} * 0.9$.
- If the dwell-time of the dominant access point decreases, the dwell-time score is decreased: $WLAN_{dwelltime} = WLAN_{dwelltime} * 0.9$.

Similar to the GSM handover score, this results in a function, which increases asymptotically towards 10 when the dwell-time of the dominant access point increases, and decreases asymptotically towards zero when the dwell-time of the dominant access point decreases. As with the GSM handover score, the constants were selected empirically. Due to the different constant values, compared to the GSM handover score, the WLAN dwell-time score changes at a slower rate; see Fig 6. The constant values reflect the update frequency of the function. For Wi-Fi, the function is updated every scan, i.e., every $1 - 2.5$ seconds. For GSM, the function is updated every time a GSM handover is detected, or after 15 seconds has elapsed since the previous update, resulting in a varying update frequency roughly between $10 - 15$ seconds.

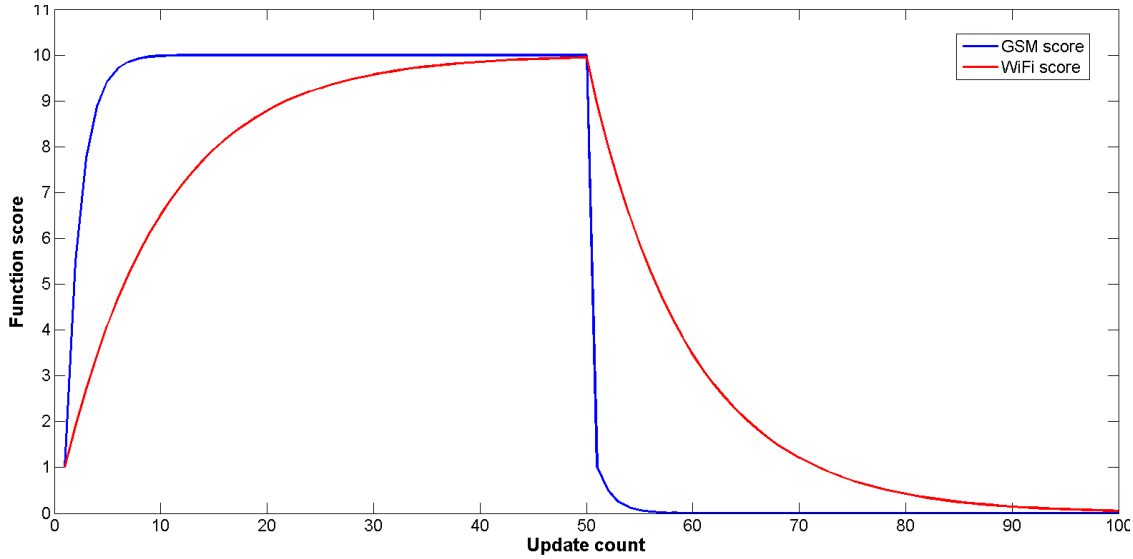

Figure 6: A comparison between the behavior of the $GSM_{handover}$ and the $WLAN_{dwelltime}$ scores. In the figure, the corresponding functions are first increased 50 times (GSM handovers are detected and the dwell-time of the dominant WiFi access point increases), after which the functions are decreased 50 times (no GSM handovers are detected and the dwell-time of the dominant WiFi access point decreases).

GPS The GPS chip on both Nexus S and Galaxy S II smartphones is the SiRF GSD4T assisted GPS chip. We poll the sensor for raw GPS data, reported in National Marine Electronics Association (NMEA) format, with a frequency of 1 Hz. From the NMEA-strings, we extract latitude, longitude, speed, number of visible satellites and the horizontal, vertical and positional Dilution of Precision (DOP). The three DOP-values measure the reliability of the GPS measurements; for more detailed information, see [YAADH00]. As a preprocessing step, we filter out unreliable measurements which have less than three visible satellites, or for which horizontal DOP values exceed 6.0. After preprocessing, we extract the speed, and three additional features introduced by Zheng et al. [ZLC+08]: heading change rate, stopping rate, and velocity change rate, defined below:

- Heading change rate measures the frequency of significant heading changes in the user's movement. A significant heading change is defined as a point where user's orientation changes more than a certain threshold H_c.
- Stopping rate measures the frequency of stops in the user's movement. A stop is defined as a point, where the user's velocity is below a certain threshold V_c.

- Velocity change rate measures the frequency of significant changes in the users velocity. A significant change is defined as a point where the user's velocity changes (in percentage) more than a certain threshold V_r.

Following the approach of Zheng et al., we define the threshold values as $H_c = 19$ degrees, $V_c = 3.4$ m/s, and $V_r = 26\%$ [ZLC+08].

Gravity and Linear acceleration sensors The gravity and linear acceleration sensors are two of the new sensing abilities in the more recent model Samsung Galaxy S2. The values of both sensors are derived from the accelerometer and as such are not separate chips. The gravity sensor outputs a three-dimensional vector indicating the direction and the magnitude of gravity in the phone's local coordinate system, while the linear acceleration sensor outputs the gravity eliminated acceleration along each of the three local axes. The sampling frequency of both sensors is identical to that of the accelerometer sensor, i.e., 100 Hz. From these two virtual sensors, we extract the same feature space as from the accelerometer. Note that the gravity estimation of the Samsung Galaxy S II phone is very similar to that of the Mizell's gravity estimation presented previously; see Figure 5. Consequently, the utility of these two sensors is only modest. Nevertheless, for completeness we also consider features from these sensors in our feature selection phase.

Magnetometer The magnetometer sensor in our use is the AKM AK8975 compass, capable of measuring the ambient magnetic field along three axes. On the Galaxy S II phones, the maximum frequency of the sensor is 90 Hz. Using the magnetometer, it is possible to determine the direction of the magnetic north, i.e., to construct a simple compass. Along with the gravity estimate this could be used to approximate the orientation of the phone. In practice, however, the sensor used in the current phone models is highly sensitive to magnetic disturbances caused by, e.g, large ferro-metal objects. Furthermore, disturbances from these sources are emphasized while within motorised vehicles. The disturbance is caused by the large metallic body of the vehicle, which induces a so called stray hard iron distortion to the magnetic field, significantly reducing the robustness of the compass. In hopes that some of the motorised transportation modalities could be detected with a specific magnetic fingerprint, we extract magnetometer features (e.g., magnitude, variation and range) that profile the magnetic field. However, none of these features passed our feature selection phase and further investigation of the magnetometer is

omitted from this study and left for future research.

Gyroscope The gyroscope is another of the new sensors only present in the Samsung Galaxy S2 phones. The chip is the STMicroelectronics L3G4200D, a three-axis MEMS gyroscope operating at up to 110 Hz frequency. The gyroscope is capable of measuring the rate of rotation around the device's local X,Y and Z axis. The sensor could be used, together with the accelerometer, to track the user's movement using dead reckoning. The sensor, however, comes with a high energy cost, consuming up to seven times more energy (155 mW compared to 21 mW; see Table 2) than the accelerometer. From the gyroscope, we extracted features that describe the angular movement of the phone, but as expected, due to the high energy cost, none of these were selected in our feature selection phase.

Rotation sensor The rotation sensor is another new sensor in Galaxy S II phones, which task is to estimates the changes in the phone's orientation. The output is a combination of angle-axis indicators; the rotation θ around axes X, Y and Z. We poll the rotation sensor at its maximum frequency rate of 90 Hz. As the functionality of the rotation sensors is very similar, we use the same feature space as for the gyroscope.

4 Hierarchical Classifier Design

The key idea in our proposed system, coined *HASMET* (Hierarchical Adaptive Sensor Management for Energy-efficient Transportation behavior monitoring), is to decompose the transportation modality detection task into subtasks. The classification proceeds from a coarse-grained classification towards a fine-grained distinction of transportation modality. At the heart of each classifier part is a hybrid classifier design, which consists of an Adaptive boosting classifier (AdaBoost) and Hidden Markov Model (HMM) classifier. While classification systems similar to ours have been previously studies, none has yet been applied on the field of transportation behavior monitoring. In the following subsections, first relevant classification techniques are outlined, followed by a detailed presentation of the HASMET feature selection scheme and classification design.

4.1 Classification

In machine learning, classification tasks are traditionally tackled with one of the two main approaches: (i) discriminative processes that focus on resolving boundary values of the classes, and (ii) generative processes that describe the underlying class distribution and state transitions [RH97]. In the following, we introduce one representative and relevant technique to our work from each approach: decision trees (discriminative) and Hidden Markov Models (generative). We also cover classifier boosting, a technique to improve any learning classifier, used in our classifier design.

4.1.1 Decision Trees

Decision trees are a well-established, versatile tool for classification tasks. One definition, presented by Rasoul and Landgrebe [SL91], describes decision trees as tree-shaped schematic diagrams designed for multi-stage decision problems. One of the main advantages of decision trees is their ability to break complex decision making problems into a series of smaller, more manageable problems (see Figure 7 for an example). An additional benefit compared to single-stage classifiers is computational efficiency of the decision trees, especially on high-dimensional feature spaces and on problems with large set of classes.

As common in classification problems, there are two phases in employing decision trees: (i) training the classifier, which in the case of decision trees is called *growing*

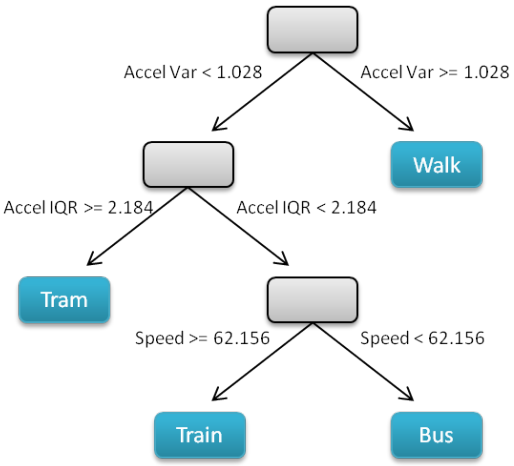

Figure 7: A simple decision tree used to classify the user's transportation modality into one of the four final classes.

the tree, and (ii) applying the classifier on unseen data in order to classify them. The latter part, classification, is relatively straight-forward and in practice can be considered simply as a chain of conditional statements. The former is a more complex task and has seen extensive research [Qui86, UBC97, Die00]. In our study, the algorithm used for training the classifier is the C4.5 algorithm [Qui93], a developed version from its predecessor, ID3 [Qui86]. A more recent, commercial version of the algorithm also exists under a name C5.0. The core of the algorithm in all versions is based on maximizing information gain, i.e., the difference in entropy. The simplified logic of the algorithm is as follows:

i) Check for base cases and stopping condition

- New class encountered.
- Information gain is zero (or below a specified threshold) for all of the features.
- All the remaining samples belong to the same class; return a leaf node containing that class.
- Stopping condition: If all the remaining nodes are leaf nodes, terminate the algorithm.

ii) For each feature F, calculate the (normalized) information gain G for splitting the set S with Feature F

$$G(S, F) = E(S) - \sum_{i=1}^{m} f_S(F_i) E(S_{F_i}),$$

where

m = number of different values of the feature F,

$f_S(n)$ is the frequency of value n in set S,

F_i is the feature F set to value defined by index i,

S_F is the set S, split with feature F_i, and

E is entropy, defined as:

$$E(S) = -\sum_{j=1}^{m} f_S(j) \log_2 f_S(j)$$

iii) Choose the feature F, with the highest information gain G.

iv) Create a new decision node containing F, that splits the remaining set S into two subsets.

v) Add the subsets obtained in (iv) as child nodes and recursively apply the algorithm on them.

While generally robust tools for classification tasks, decision trees also have some issues which need to be considered. The primary concern is overfitting the data, which occurs when noise or random variation is interpreted as identifiers for a class. Overfitted classifier typically fails to generalize, resulting in poor performance on unseen samples. In case of decision trees, overfitting can be alleviated to some extent by performing pre-, and post-pruning on the tree (i.e., reducing the size and complexity of the decision tree). Pre-pruning is performed while growing the tree, for example by setting a criteria for the amount of observations to create a new node, and post-pruning is performed after the tree is grown by reducing the size of the decision tree. While pruning reduces the tree's predictive strength on the training data, it results in more robust classification on unseen data.

4.1.2 Adaptive Boosting

Boosting is a general technique for improving the accuracy of any learning algorithm [FS99b]. The basic idea in boosting is to iteratively learn weak classifiers that focus on different subsets of the training data, and combine these classifiers into one strong classifier. Adaptive boosting (AdaBoost), introduced by Freund and Schapire in [FS95], extends the idea of boosting by tuning to the problematic samples misclassified by previous classifiers. Specifically, AdaBoost operates by assigning each sample in the training data a weight that determines the importance of the sample. Over a series of rounds $t = 1, \ldots, T$, classifiers that minimize classification error on the weighted data are learned. After each round, the weights of the samples are

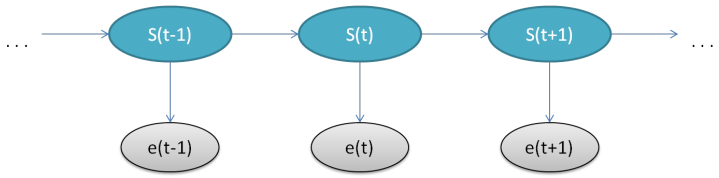

Figure 8: A sample HMM-diagram visualizing transitions between states.

re-evaluated to assign higher priority to samples that are misclassified. Note that features that are selected by the weak learners provide us with an automatic way to identify the most relevant features for the final classifier design.

4.1.3 Hidden Markov Models

Hidden Markov Models (HMM) are another widely used machine learning tool capable of classification tasks. In broader terms, HMM can be considered a statistical Markovian method for modeling series of underlying hidden states, with an observable sequence of emitted values (see Figure 8). The hidden states have initial probability distribution P_i, state transition probability distribution P_t and emission probability distribution P_e. If the three probability distributions are known, the most likely state or sequence of states corresponding to the observed emission(s) can be efficiently solved. In addition, the HMM has a specific order, which corresponds to the length of the history, i.e., the number of previous states, to consider in calculating the posterior probability distribution. After being rediscovered by engineering sciences in the late 1980's [Rab89], HMMs have been adopted on various fields and problems. In particular, HMMs are effective in solving temporal pattern recognition problems and as such have been applied widely in the activity recognition field [TIL02, WPP+07, AY09].

In terms of classification approach, HMM is a generative classifier. Generative classifiers approach the classification task by learning a model of the underlying joint probability $p(s, c)$, where s denotes samples and c denotes the class. The predicted class is the one yielding the highest probability $p(c|s)$. In terms of classification accuracy, discriminative classifiers (such as decision trees) typically outperform generative classifiers [NJ02]. This is due to the discriminative classifiers modeling the class conditional probability $p(c|s)$ directly, while generative classifiers, in a sense, try to solve a more general problem in estimating the underlying joint distribution $p(s, c)$. However, as these methods have complimentary strengths, the most efficient technique is typically one that combines a generative classifier with a discriminative

classifier [JH98].

4.1.4 Hybrid Classifiers

An effective approach in activity recognition related classification tasks is to construct a multi-stage hybrid classifier consisting of a combination of classifiers. We describe here one such classifier combining a decision tree with a Hidden Markov Model, a method used in several previous studies [RMB+10,LCK+05]. The efficiency of this hybrid design owes to the complementary strengths of the two classification methods:

- The decision tree is efficient in processing the raw features and finding boundary values for different classes, i.e., performs hard classification.
- HMM incorporates domain specific knowledge into the classifier in the form of state transition probabilities, i.e., smooths out noise and prevents unlikely state transitions based on temporal knowledge of the previous state(s).

An DT+HMM hybrid classifier (see Figure 9 for an example) can be constructed with either of the following approaches: i) dedicate a separate DT+HMM pair for each class, or ii) model the classes as hidden states of a single HMM. In case of transportation behavior monitoring, the latter has been shown to be a both simpler and more efficient approach [RMB+10], and consequently we will focus on it only. The classifiers are ordered so that the initial classification is performed by the decision tree. From the initial classification phase, a posterior probability distribution P_{dt} is obtained. In the second phase, the HMM classifier is trained with P_{dt} (i.e., P_{dt} corresponds to P_e). Transition probabilities P_t can be either tuned manually by an expert, or derived from the frequencies of state transitions in the data. The initial probability distributionP_i reflects the initial situation and is typically set to a uniform

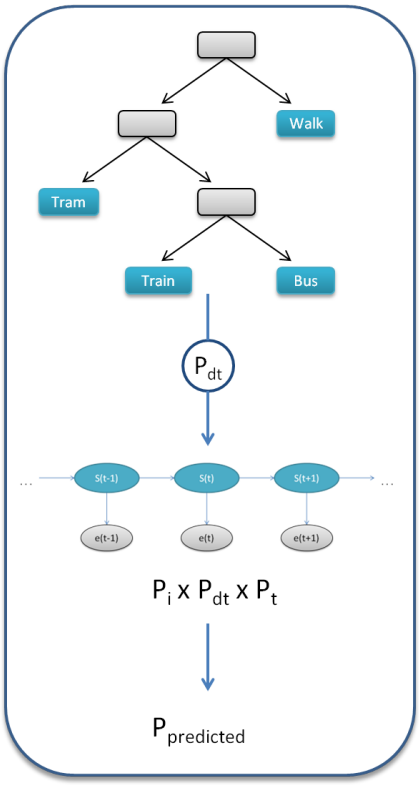

Figure 9: Diagram of an example Hybrid Classifier control flow.

distribution. The predicted class is given by calculating P_i x P_{dt} x P_t, and choosing the class with the highest probability.

4.1.5 HASMET Classifier Design

The HASMET classifier design, used within all the classifier parts, consist of a two-stage classifier, similar to the one outlined in the previous subsections. The initial classification is performed by an adaptive boosting classifier, consisting of an ensemble of low-depth decision trees. While the AdaBoost algorithm is generally robust against over-fitting, mislabeled data or other outliers could result in decreased performance. To reduce the risk of over-fitting, we use a gentle variant of AdaBoost, which puts less emphasis on outliers than the original algorithm [FHT00]. The number of boosting rounds T was determined using the scree-criterion, i.e., we plot the classification error for varying values of T and select a suitable value of T, which balances between the classifier accuracy and classifier complexity. In order to retain classifier simplicity and to prevent over-fitting to training data, we opted for the minimal T value (currently 25), after which further increasing the value of T results only in small gain in accuracy.

From the initial classifiers' margin values, we derive a probability distribution and pass this to the second stage classifier, for which we use a first-order discrete HMM. The characteristics that we seek to capture with the HMM are the continuous nature of the transportation behavior (i.e., smooth the predictions over time) and the probabilities of transitions between different states. The state-transition probabilities are derived from our dataset as likelihood ratios between consecutive transportation modalities, with unlikely transitions (e.g., from biking to motorised or running) set to zero probability.

4.2 Sensor and Feature Selection

In activity recognition, features are the measurable attributes of an observed phenomenon. Finding the correct features that are able to identify each activity has a significant impact for the whole system. As described above in Section 4.1.2, employing boosting provides us an automatic method for selecting a feature set which best distinguishes between different classes. To provide an optimal trade-off between energy efficiency and detection accuracy, we have modified the loss function used in AdaBoost for feature selection to also consider the energy cost associated with

the features. Specifically, we first weight the score of each feature, given by the original AdaBoost algorithm, with a value relative to the respective sensor's power consumption. Next, we find the features which provide similar detection accuracy, i.e., which are within a small range of each other, and choose the feature which results in the smallest increase to the total energy cost of the classification task. As a result, the features from sensors with low power consumption, or sensors that have already been included to the classifier design, are prioritized over the features from sensors with high power consumption.

As the result from applying the modified AdaBoost algorithm on our initial feature space containing features from all eight sensors, described in 3.3, features were selected from the following four sensors: accelerometer, GPS, GSM and Wi-Fi. As excluding the high consuming GPS from the classifier is one of our research goals, we applied the algorithm on two sensor sets, one including and the other excluding GPS. This resulted in two classifier designs, one where accelerometer, GSM and Wi-Fi are used for detection, and one where GPS is additionally used. In the experiments we refer to these two versions as HASMET and HASMET$_G$, respectively.

4.2.1 Feature examination

From the initial feature space of 172 features, 24 features were selected for different stages of classification. Here we give a description for each of the features utilized by HASMET.

Total Acceleration presents the combined magnitude of the kinematic forces applied on the smartphone. Eight features were selected from this representation: variance, minimum, maximum, range, interquartile range, kurtosis, wavelet entropy and FFT direct component. Variance measures the fluctuation of the accelerometer magnitude and is effective in separating between activities with different intensity and pace of movement (e.g., walking and running). Minimum and maximum measure the largest and smallest values detected within a window and are usable in separating between modalities with similar variation. Range measures the difference between maximum and minimum values, with similar utility as maximum and minimum. Interquartile range functions as the range, with the exception that it eliminates the highest and lowest quartiles, and as such is often more robust against noise and sporadic spikes in the accelerometer measurements. Kurtosis measures the peakedness of the values within a window and its effectiveness is based on find-

ing specific patterns from motorised activities. Wavelet entropy measures similar attributes as kurtosis, with the difference that it examines the frequency representation of the accelerometer values. As with kurtosis, wavelet entropy functions by distinguishing between activities with similar energy levels based on different activity patters. Finally, the FFT direct component characterizes the frequency signal of an activity and is able to distinguish between activities with different intensities of physical movement.

Horizontal Acceleration presents the acceleration along the horizontal plane, i.e., forward, backward and lateral movement. From this representation, six features were selected: variance, range, interquartile range, minimum, mean cross-correlation and integral. The first four features are analogous to those described above, with the exception that emphasis is placed on transportation modalities which can be best identified from their horizontal acceleration. The first of the two remaining features, mean cross-correlation is the average of cross correlations between each pair of axes (i.e., {x,y}, {x,z} and {y,z}), and is usable for identifying movement which distributes unevenly along the axes, such as acceleration and deacceleration periods of motorised vehicles. Integral measures the accumulated acceleration over the window, which can operate as a proxy for momentary speed gain. Similarly to the mean cross-correlation, the integral of the horizontal acceleration is effective in capturing acceleration which is directed primarily to one orientation.

Vertical Acceleration measures the upwards and downwards oriented acceleration. A total of six features were selected from this representation: variance, minimum, maximum, wavelet entropy, mean cross-correlation and wavelet magnitude. The vertical features have two important functions. First, they help to identify between pedestrian modalities, in particular between bicycle and walking as only the latter has significant vertical movement. Second, the vibrations and shakes associated with motorised transportation modalities are effectively captured by the vertical features. The first five features selected from the vertical representation are described above. The remaining feature, wavelet magnitude, measures the total energy of the frequency representation of the data window, and much like the FFT direct component, identifies activities based on different levels of physical intensity.

GSM features measure the changes in the user's GSM environment, which correlates with the user's movement between places. From the GSM feature space

only a single feature, the time-interpolated handover score, was selected. This feature, however, along with the Wi-Fi features, is vital at robustly discerning between low-intensity kinematic movement and motorised transportation modalities. As presented in Section 3.3, the time-interpolated handover score initiates from a value of one, and if no GSM handovers are detected, decreases over time towards zero. In case a handover is detected, the score is progressively increased towards 10. To prevent increasing the score in a common ping-pong handover situation, where the primary GSM cell tower fluctuates between two towers [MEBH08], we store the cell identifiers from the previous 30 seconds and filter out handovers to any of the cells seen within the time frame.

Wi-Fi features are used to detect changes in user's Wi-Fi signal environment, i.e., to distinguish between moving within a place and moving between places. Two features from the Wi-Fi feature space were selected. The first is the time-interpolated dwell-time score, presented in Section 3.3. The score function behaves much like its GSM counterpart, increasing when the dominant access point remains visible and decreases when the access point is no longer visible. The second selected Wi-Fi feature is the unfiltered dwell-time, which measures the duration that the strongest visible access point has been visible. As discussed in Section 3.3, the unfiltered access point list contains all visible access points, including the universally administered ones.

4.3 HASMET Architecture

The core of HASMET consists of four classifiers which are organized into a hierarchy; see Figure 10 for illustration. At the root of the classifier hierarchy is a *kinematic motion classifier*, which uses an accelerometer to perform a coarse-grained distinction between pedestrian and other modalities. When the kinematic motion classifier detects substantial physical movement, a *pedestrian classifier* is consulted, otherwise the process progresses to a *stationary classifier*. The pedestrian classifier uses the accelerometer to separate between walking, biking and running. The stationary classifier, on the other hand, uses a combination of accelerometer, GSM and Wi-Fi to determine whether the user is stationary or in a motorised transport. When motorised transportation is detected, the classification proceeds to a *motorised classifier*, which is responsible for classifying the current transportation activity into one of five modalities: bus, train, metro, tram or car.

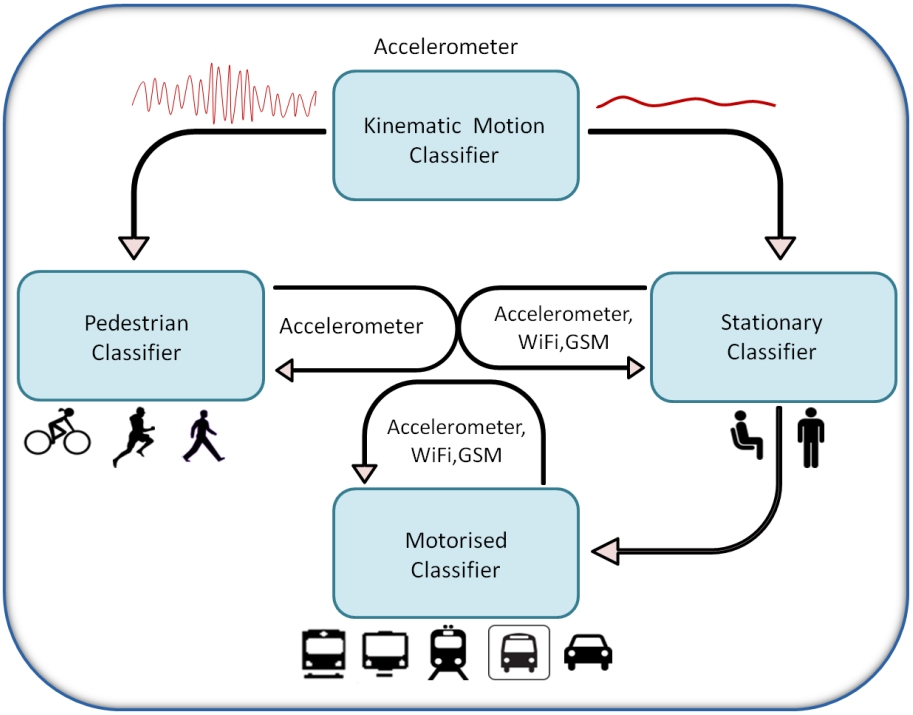

Figure 10: Overview of the the classifier decomposition of HASMET and the dependencies between the classifier parts.

Changes in transportation behavior typically occur infrequently and each activity has a duration of several minutes. HASMET utilizes this observation to improve the robustness of the transportation behavior monitoring. Specifically, HASMET follows a two-stage control flow approach where the first step is to consult the classifier that was responsible for the most recent modality estimate. If no changes in modality are detected, the process is halted. Otherwise the classification is propagated back to the parent classifier.

4.3.1 Kinematic Motion Classifier

The kinematic motion classifier utilizes accelerometer measurements to distinguish between pedestrian and other modalities. The features that were selected for the kinematic motion classifier are a set of standard statistical metrics that characterize the overall volume of kinematic movement. For a full list of the selected features, see Table 4. Note that features from all three accelerometer representations were selected by the AdaBoost algorithm, indicating that our approach with the accelerometer feature design is an efficient one. The accuracy of the kinematic motion classifier is close to 99%, which means that it can robustly determine the subsequent

Sensor	Features
Total Acceleration	Variance, Maximum, Interquartile Range
Horizontal Acceleration	Variance, Range, Interquartile Range
Vertical Acceleration	Minimum, Standard Deviation

Table 4: Sensors and features used in the kinematic motion classifier.

classifier to apply.

4.3.2 Pedestrian Classifier

The pedestrian classifier is responsible for distinguishing between walking, running and biking. The pedestrian classifier relies only on the accelerometer. The final set of features that were selected corresponds to the feature list used with the kinematic motion classifier; see Table 4. As discussed in Section 3.1, we were unable to collect a sufficient amount of measurements for biking or running due to weather conditions and consequently the accuracy of the pedestrian classifier is not considered as part of the evaluation. Previous work [BI04, RMB+10], as well as small scale experiments that we have conducted with data from a single individual indicate that these modalities can be easily identified with over 95% accuracy.

4.3.3 Stationary Classifier

The stationary classifier utilizes measurements from the accelerometer, GSM and Wi-Fi to distinguish between stationary and motorised modalities. Compared to the kinematic and pedestrian classifiers, the final feature space of the stationary classifier is significantly more diverse. The feature space encompasses statistical, time-domain and frequency-domain features from the accelerometer, as well as features from GSM and Wi-Fi sensors; see Table 5. The final stationary classifier design achieves an accuracy of approximately 85%. The more complex feature space, along with the decreased accuracy compared to the previous two tasks, indicate the difficulty of achieving this task using only the coarse-grained mobility information that GSM and Wi-Fi provide. Nevertheless, the accuracy of the stationary classifier still suffice to

provide robust continuous detection of transportation modality, even in the presence of periods where stationarity is interleaved with motorised transportation.

Sensor	Features
Total Acceleration	Variance, Range, Wavelet Entropy, Kurtosis, FFT Direct Component, Interquartile Range
Horizontal Acceleration	Variance, Integral (Velocity), Interquartile Range
Vertical Acceleration	Wavelet Entropy, Mean Cross-Correlation
GSM	Time-interpolated Handover score
Wi-Fi	Time-interpolated Dwelltime score (filtered), Dwell-Time (unfiltered)

Table 5: Sensors and features used in the stationarity detection.

4.3.4 Motorised Classifier

The motorised classifier is responsible for distinguishing between car, bus, train, tram and metro. The final set of features that were selected for the classifier are similar to those of the stationary classifier. For a full list of selected features, see Table 6. As movement patterns between different motorised modalities are very similar, this is the most challenging task in our design. As such, we have made two additional extensions to the HASMET classification scheme to improve the robustness and accuracy of the motorised detection. First, whenever the motorised classifier fails to detect the current modality, but the stationary classifier determines the user to be stationary, we assume that the user is within motorised transport that has stopped and the parent classifier is set to be the motorised classifier (i.e., the classifier that is consulted once the stationary period ends is the motorised classifier). This scheme reduces false detections from stops, such as picking up or dropping passengers or stopping at traffic lights. Second, to improve the robustness of detecting between the different motorised modalities, the current motorised transportation modality is estimated using a majority vote of the motorised modalities that have been predicted within the last 60 windows. As each window is roughly 1.2 seconds long, and the windows overlap by 50%, this corresponds to roughly 36 seconds. This

step significantly improves the robustness of the motorised detection. Note that only windows classified as motorised transport are taken into account in the voting, which enables our system to rapidly react to changes in transportation modality. This design is able to determine the correct modality with approximately 70% accuracy. Moreover, most errors result from mixing between train, tram and metro. In applications, these errors could be alleviated, e.g., by fusing in information about transportation routes or by considering more elaborate features; see Section 6.

Sensor	**Features**
Total Acceleration	Variance, Interquartile Range, Kurtosis, Wavelet Entropy, Maximum
Horizontal Acceleration	Variance, Mean Cross-Correlation, Minimum, Interquartile Range
Vertical Acceleration	Wavelet Entropy, Minimum, Wavelet Magnitude, Variance, Mean Cross-Correlation, Maximum
GSM	Time-interpolated Handover score
Wi-Fi	Time-interpolated Dwelltime score (filtered), Dwell-Time (unfiltered)

Table 6: Sensors and features used in the motorised classifier.

5 Evaluation

In this chapter we investigate the performance of HASMET. In the evaluation we consider four different aspects of the detection task: detection accuracy, detection robustness and latency, detection generalizability, and energy efficiency. For evaluating detection accuracy, we use standard classification metrics, i.e., precision, recall and F-score, as these give a good overview of the classifiers' overall capability to identify the correct transportation modality. For evaluating robustness and latency, we use event and frame-based metrics introduced by Ward et al. [WLT06, WLG11]. Detection generalization is evaluated with cross-user and cross-placement evaluations. Finally, to evaluate the energy efficiency of the different approaches we calculate their total energy consumption over our testing set. For calculating the energy consumption, we use the empirical power models presented in Section 3.2. For HASMET, we additionally calculate the energy consumption of the different classifier parts separately. All the evaluations, except for the detection generalization evaluation, were performed using the datasets summarized in Table 1.

Baselines As baselines for our evaluation, we compare HASMET with two existing methods for transportation behavior monitoring. The primary baseline is the approach of Reddy et al. [RMB+10], as this presents the current state-of-the-art solution in transportation behavior monitoring on mobile devices. The approach utilizes a combination of accelerometer and GPS features. From the accelerometer, the approach extracts four features: variance and three FFT components of the L_2 norm corresponding to the frequencies $1-3$Hz. From GPS only speed is used as a feature. Classification is performed with a hybrid classifier consisting of a decision tree and a first order discrete HMM classifier. Since the baseline method utilizes GPS, and considering that many of our target applications would already employ GPS for other functionalities, we perform the comparison both against HASMET and HASMET$_G$. As the secondary baseline we use the approach of Mun et al. [MEBH08], as it relies on similar features from GSM and Wi-Fi as our approach. From the GSM measurements, the approach of Mun et al. extracts the dwell-time in the primary cell, and the number of unique cells seen within a window. Both features are calculated over a two minute window. From the Wi-Fi measurements, two similar features are extracted: the dwell-time of the dominant access point and its signal strength variance, both calculated over a window of 40 seconds.

5.1 Detection Accuracy

We begin the evaluation by considering the classifiers' overall ability to detect the correct transportation modality. For this purpose, we employ three standard set-based metrics: precision, recall and F-score. Precision measures the classifiers' ability to avoid false classification, while recall measures the classifiers' ability to detect the right transportation modality. The F-score is the harmonic mean of the former two. Specifically, we consider the following metrics:

$$Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

$$F1_{score} = 2 * \frac{recall*precision}{recall+precision} = \frac{2TP}{2TP+FN+FP}$$

Where TP = True Positive, FP = False Positive and FN = False Negative

The results of the accuracy evaluation along with a comparison with the baselines are presented in Table 7. Overall, HASMET achieves on average 80% precision and slightly lower recall at around 77%. The GPS augmented version HASMET$_G$ achieves increased performance, reaching around 84% on both precision and recall. Both of the HASMET variants offer improved detection accuracies over the baseline approaches. The baseline of Reddy et al. achieves average precision and recall roughly at 70% and 68% respectively. Note that while Reddy et al. reported average precision and recall values close to 90% [RMB+10], these results were achieved by combining all the motorised transportation modalities into a single class. In case we merge all the motorised modalities into a single class, the results are analogous to those reported by Reddy et al., and HASMET still retains better performance. The other baseline from Mun et al. achieves average precision and recall both around 44%. The notable differences to the reported results from Mun et al. [MEBH08] are caused by a more demanding testing scenario used in our evaluation. The evaluation carried out by Mun et al. included long periods of dwelling (e.g., having a dinner), while our evaluations focuses exclusively on the active part of moving between places.

	Precision				Recall				F-Score			
	Mun	Reddy	HASMET	HASMET$_G$	Mun	Reddy	HASMET	HASMET$_G$	Mun	Reddy	HASMET	HASMET$_G$
Still	50.8	87.0	78.5	83.8	38.2	76.7	90.5	88.9	43.6	81.5	84.1	86.3
Walk	50.7	97.5	98.3	98.4	49.7	98.5	98.8	98.8	50.2	98.0	98.6	98.6
Bus	20.3	64.2	67.3	70.3	17.8	54.2	72.1	87.8	19.0	58.8	69.6	78.1
Train	42.2	40.1	69.8	77.4	31.6	12.1	54.8	60.8	36.1	18.7	61.4	68.1
Metro	46.2	47.8	77.7	81.0	45.3	62.2	54.0	72.1	45.7	54.2	63.7	76.3
Tram	34.6	57.2	74.5	83.3	50.0	78.9	76.2	80.8	40.1	66.4	75.3	82.0
Car	62.2	95.9	92.6	99.5	71.2	92.4	91.6	96.7	66.4	94.1	92.1	98.1
Mean	43.8	70.1	79.8	84.8	43.4	67.9	76.9	83.7	43.1	67.4	77.8	83.9

Table 7: Classification accuracy of HASMET, with and without GPS, and comparison against the baseline systems of Reddy et al. [RMB+10] and Mun et al. [MEBH08].

	Still	Walk	Bus	Train	Metro	Tram	Car
Still	16715	131	507	277	408	750	14
Walk	59	16541	46	20	33	38	8
Bus	236	47	5224	118	0	325	0
Train	461	25	812	3221	246	503	31
Metro	904	24	325	526	5681	416	0
Tram	1571	31	166	1	644	10150	0
Car	2	12	349	0	0	0	10702

Table 8: Confusion matrix of the classification performed by HASMET.

Inspecting the performance of the different classifier components in HASMET reveals, that the kinematic classifier performs very reliably at around 98% precision, and 99% recall values. Observing that for the kinematic modalities the accuracies achieved with the approach of Reddy et al. are also very high, we conclude that this task is relatively simple and solely an accelerometer is sufficient for this task. The stationary classifier performs slightly less reliably with 90% precision and 79% recall, but still at a level that is acceptable for most real-world applications. Not surprisingly, the main challenges emerge from correctly predicting the motorised transportation modality. The motorised classifier achieves average precision and recall of around 76% and 70% respectively. Most of the false classifications result from difficulties with distinguishing between the different public transportation modalities. This part of the classification task is, however, also where HASMET outperforms the primary baseline approach of Reddy et al. most clearly. This is due to the GPS performing less reliably inside motorised vehicles, and the more intelligent accelerometer feature design we use in HASMET. As correctly recognizing the public transportation modalities is one of the most interesting applications domain for mobile transportation behavior monitoring, improving the performance of the motorised classifier is one of the main objectives of our future work; see Section 6.

5.2 Detection Robustness and Latency

In evaluating a continuous activity recognition system, the metrics considered in the previous section give a incomplete view of the system's suitability for applications. In particular, considering only direct frame-by-frame metrics, such as precision and recall, ignores the robustness of the detection, i.e., the fragmentation of the de-

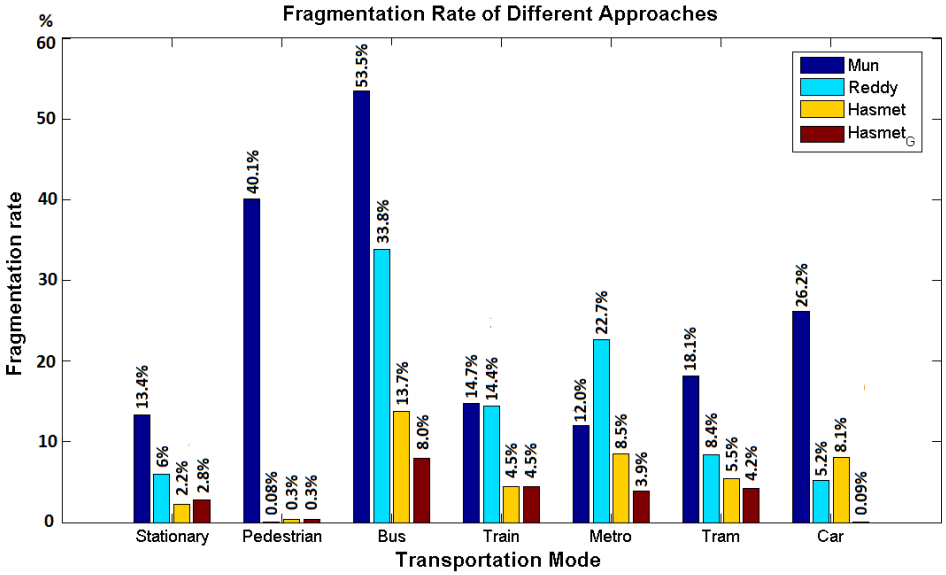

Figure 11: Fragmentation rate of of the different approaches.

tection, and offers no view to the systems detection latency. To incorporate these aspects into our evaluation, we employ the fragmentation rate and the underfill rate proposed by Ward et al. [WLG11]. The fragmentation rate expresses the fraction of all ground truths events that the classifier misinterprets as multiple events, while the underfill rate measures the delay in detecting the correct modality, when a transition from one modality to another occurs.

The evaluation results of these metrics are presented in Figure 11 and Figure 12. Overall, compared to both of the baselines, HASMET provides improved robustness in the transportation behavior monitoring with lower fragmentation rate in all the other modalities, except for driving and walking, for which HASMET and Reddy et al. perform approximately equally. For walking, the fragmentation rate of HASMET is minimal at around 0.6%. For driving, the fragmentation rate is also low at around 8.1%. HASMET$_G$ further increases the robustness, providing lower fragmentation rate for all the other modalities, except for walking, for which the HASMET$_G$ has a negligible 0.6% fragmentation rate. Similar to the detection accuracy, the main improvements in robustness over the baselines result from the better robustness in detecting between the different motorised transportation modalities.

Inspecting the results from the other event metric, underfill rate, we first observe that all four evaluated methods are efficient at detecting transitions from one modality to another. Overall, compared to the baseline systems, HASMET provides lower latency to the detection, evident from the smaller underfill rates; see Figure 12.

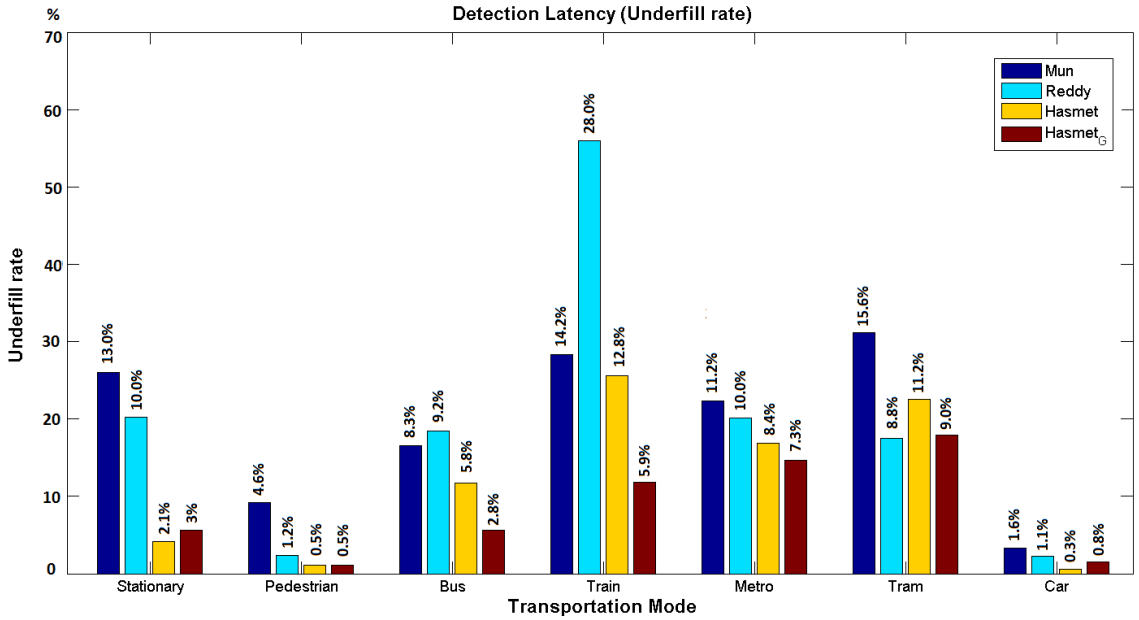

Figure 12: Underfill rate of the of the different approaches.

Compared to the approach of Mun et al., HASMET induces smaller latency for all the modalities. This is due to the approach of Mun et al. relying on features extracted over windows with longer duration. Compared to the approach of Reddy et al. HASMET induces lower latency for all the other modalities, except for tram, for which HASMET has an underfill rate of 11.2% and Reddy et al. have an underfill rate of 8.8%. Together with the higher underfill rate for stationarity and higher fragmentation rate in detecting motorised modalities, the results suggest that the reliance on the speed given by the GPS causes errors during stationary periods in motorised transportation. HASMET overcomes this problem through improved feature design and by approximating speed with alternative sensors that, unlike the GPS sensor, are able to operate reliably also when obstructed.

5.3 Detection Generalizability

In a continuous sensing system, where the system is running in the background of a mobile phone for extended periods of time, user convenience and freedom from artificial restrictions are of central importance. To ensure that our system provides cross-user compatibility and is not sensitive to varying phone placement or orientation, we have conducted two sets of separate experiments: cross-user, and cross-placement evaluations. The cross-user evaluation was conducted by perform-

	Precision			Recall			F-Score		
	Mun	Reddy	HMT	Mun	Reddy	HMT	Mun	Reddy	HMT
Still	57.5	85.9	73.3	38.9	81.1	91.1	45.8	83.3	81.1
Walk	43.0	96.9	97.8	55.7	98.5	98.9	48.2	97.7	98.3
Bus	33.6	75.9	77.5	31.3	53.7	82.1	31.8	62.7	78.6
Train	59.8	41.0	80.1	58.7	11.4	67.8	57.2	16.8	72.5
Metro	49.6	43.0	68.9	53.0	54.4	62.8	50.1	46.9	65.7
Tram	42.3	56.5	71.1	50.0	78.6	52.2	45.4	65.4	60.1
Mean	47.6	66.5	78.1	48.0	62.9	72.5	46.4	62.1	76.1

Table 9: Results from the cross-user experiments, along with comparison to the baseline systems. The presented values are the mean values over all the six cross-user tests.

ing a series of leave-one-out cross validation tests (six in total), i.e., by training the classifier with the data from all users, excluding one, and testing the classifier with the data from the user not present in the training set. The mean precision, recall and F-score of the different transportation modalities, along with the baselines, are presented in the Table 9. From the results, we observe a small decrease, ranging from two to five percent, in accuracies for both HASMET and the approach of Reddy et al. As this decrease is not present in the results of the approach of Mun et al., which has in fact improved around 5%, we assume that the decrease originates from the accelerometer features, only present in HASMET and the approach of Reddy et al. This is in accordance with the notion that the kinematic features are more susceptible to user variations than those that measure changes in the user's environment [LCB06]. Overall, we conclude that the classifiers' accuracies suffer only a subtle decrease in cross-user evaluations and that all three evaluated systems generalize well over a variety of users.

The cross-placement evaluation was conducted similarly, i.e., by training the classifier with data from two of the three placements, and testing the classifier with data from the left-out placement. The procedure was repeated for all three possible combinations of placements. The results of this evaluation are presented in Table 10. The results are similar to those of the cross-user evaluation, with a small decrease in accuracies, particularly for the motorised modalities. Overall, the accuracies remain on a comparative level to those of the previous evaluations, and we conclude that all evaluated systems also generalize over varying phone placements.

	Precision			Recall			F-Score		
	Mun	Reddy	HMT	Mun	Reddy	HMT	Mun	Reddy	HMT
Still	58.6	85.8	74.2	39.8	82.6	91.7	46.8	84.1	82.0
Walk	43.8	96.8	97.8	54.2	98.8	98.9	48.2	97.8	98.4
Bus	34.7	76.7	74.9	29.2	51.1	78.7	30.1	60.0	76.1
Train	51.3	53.1	78.6	66.4	9.2	68.1	54.8	15.4	70.7
Metro	50.1	42.0	46.3	50.1	49.8	53.7	50.4	42.9	49.7
Tram	43.0	54.4	70.7	51.2	75.5	58.3	47.0	62.7	63.2
Mean	46.9	68.1	73.8	48.7	61.2	74.9	46.1	60.5	73.4

Table 10: Classification accuracy results from the cross-placement experiments, along with comparison to the baseline systems. The presented values are the mean values over all three cross-placement tests.

Note that both of the previous evaluations were performed solely with the data collected from the scenario, where no driving sections were present. To verify that detecting driving also generalizes over different users, we have conducted an additional experiment using the data from the everyday data collection. The classifier was trained using data from one user and tested with data from another user. The achieved precision and recall are 92.4% and 90.9% respectively, which indicates cross-user generalization for this modality as well. Evaluating the cross-placement generalization for the driving modality is omitted from this study, as we currently have only limited amounts of data from different placements for this modality. This is due to that the users typically remove the phone from their pocket before driving, or in case that the phone was held in a jacket pocket, remove their jacket before driving.

5.4 Power Consumption

While operating on a user's mobile phone, it is of significant importance to keep the application's resource demands at minimum. While the computational power and the available memory of mobile phones has witnessed a rapid increase, the most limiting resource today is the phone's battery energy. As such, our aim is to provide accurate and continuous transportation behavior monitoring, while simultaneously minimizing the detection's impact on the operational time of the phone. In order to evaluate the system's energy efficiency, we utilize the empirical power models de-

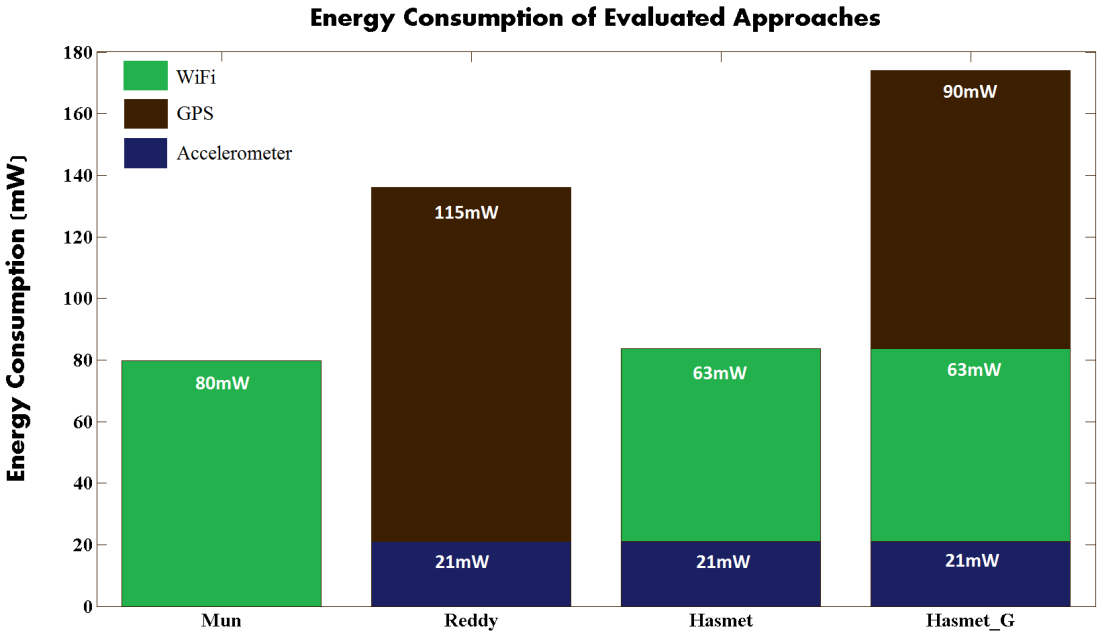

Figure 13: Average power consumption of the different approaches in mW.

scribed in Section 3.2 for estimating the average power consumption of the different approaches.

In our energy-efficiency evaluation, we consider four different systems: HASMET, HASMET$_G$, the approach of Reddy et al. and the approach of Mun et al. The results, along with the decomposition of the different sensors, are presented in Figure 13. Note that we have omitted the power consumption of two aspects of the transportation monitoring: the energy cost associated with polling the GSM cell information, and the power consumption resulting from the CPU load. The former was omitted due to that the phone is already continuously polling this information and we merely read the values as they are received. The latter was omitted due to the relatively light CPU load of the detection, and due to the CPU loads negligible energy footprint compared to the energy consumed by the sensors.

The dataset, detailed in Section 3.1, used for testing consists of 13 hours of various transportation modalities. The results demonstrate that HASMET consumes roughly $3,935$ J over our test set, i.e., it has an average power consumption of 84 mW. Compared to the approach of Reddy et al., which consumes in total $6,394$ J and on average 136 mW, our approach reduces the power consumption by approximately 40%. The alternative design, HASMET$_G$, increases the power consumption compared to approach of Reddy et al. by approximately 20%, having a total con-

sumption of 8,177 J, which corresponds to a 174 mW average power consumption. The approach of Mun et al. uses only Wi-Fi and GSM and as such it has the lowest power consumption at average of 80 mW. This small, approximately 5%, reduction in power consumption, however, is quite modest compared to the significant accuracy trade-off evident from the previous evaluations.

With the decomposition of the classification task, the power consumption of the transportation behavior monitoring varies depending on the classifier part used. At the root of the hierarchical classification we use the kinematic classifier, which relies purely on the accelerometer, resulting in an average power consumption of 21 mW. The pedestrian classification is similarly executed by only employing the accelerometer, resulting in the same average power consumption of 21 mW. For the stationary and the motorised classifiers, we additionally require GSM and Wi-Fi sensors, which increases the power consumption up to 101 mW. The GPS augmented version of these two classifiers, used only by the HASMET$_G$, results in the highest power consumption of 206 mW. Note that the values referred above reflect the specific scanning frequencies used in this study; see Section 3.3. Further energy efficiency is achievable by investigating the relation between sampling frequency and classification accuracy, an aspect of our future work.

6 Summary and Conclusions

In this thesis, we have presented HASMET, a hierarchical system designed for energy-efficient transportation behavior monitoring on mobile phones. The key idea of the system is to decompose the monitoring task into smaller subtasks and only employ the necessary sensors for each subtask. Our hierarchical approach, along with intelligent feature design and an effective hybrid AdaBoost+HMM classification scheme, enables detection of fine-grained transportation modalities without the power consuming GPS sensor, relying instead on less power consuming sensors, i.e., accelerometer, GSM and Wi-Fi. The transportation modalities HASMET is able to detect cover all the relevant locomotion types present at the target area (Helsinki), including the detection of the different motorised transportation modalities, i.e., bus, train, metro, tram and car. Experiments conducted with a total of 33 hours of data, collected from seven users and three phone placements, demonstrate that compared to the state-of-art, HASMET achieves significant improvements in energy-efficiency, while also improves the detection accuracy and robustness.

In terms of future work, we have already integrated a mobile version of the HASMET as a part of a persuasive mobile application that encourages ecological transportation behavior. A long-term field study will be carried out within this research, offering insights into the system's performance. This will also give us an opportunity to investigate how the detection errors of transportation behavior monitoring affect the user experience. The long-term study should also provide us with more data from running and biking modalities, offering a possibility to conduct a thorough evaluation of the pedestrian classifier.

The biggest challenges with the current system relate to distinguishing between different public transportation modalities. One promising approach is to improve the motorised classifiers' performance by extracting more information from the accelerometer. By further developing our gravity estimation algorithm, possibly along with the magnetometer to estimate the exact phone orientation, we can construct more accurate acceleration profiles for the different (motorised) transportation modalities. Features that could be derived from this approach include the intensity and temporal distribution of the acceleration periods of motorised vehicles, usable in distinguishing between public transportation vehicles; see Figure 14 for an illustration. The data presented in the figure is the gravity eliminated horizontal acceleration, recorded using a mobile phone embedded accelerometer. Note that the figure represents the current state of our algorithm development, and we expect to

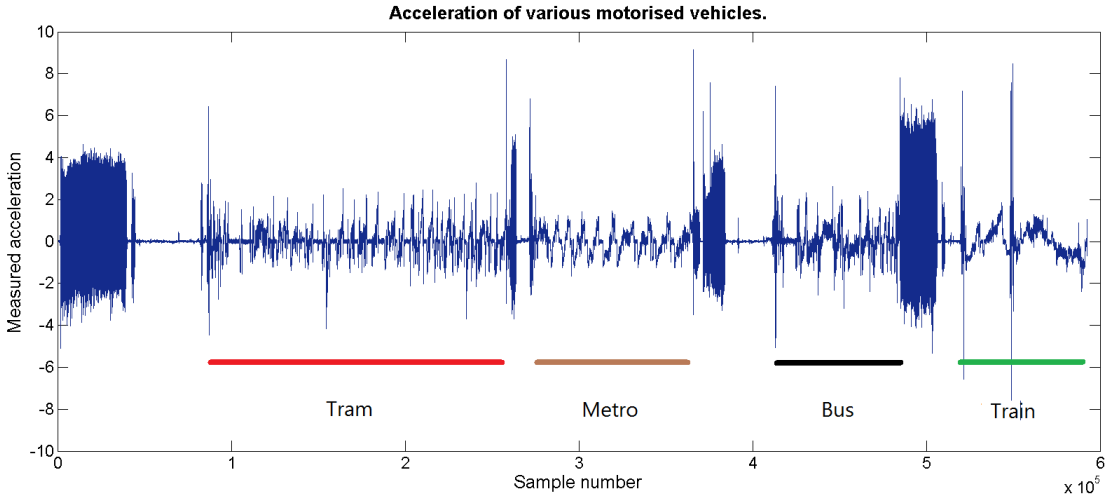

Figure 14: Gravity eliminated horizontal projection of the acceleration for different motorised modalities.

achieve more accurate acceleration profiles with further progress. In addition to the accelerometer, we will also continue exploring the possible uses of the other sensors present on the modern mobile phones. As many of the public transportation vehicles have distinguishing audio space, we will investigate the possible use of audio features extracted from the phone's microphone. Additionally, we will experiment with the light and proximity sensors in order to detect periods of user interaction, which are a significant source of detection errors in transportation behavior monitoring.

While HASMET already provides substantial reduction in power consumption, the reported reduction is mainly due to using sensors with lower energy consumption. Further advances in energy efficiency can be easily achieved by reducing the scanning frequency of the Wi-Fi sensor and by implementing duty cycling on the accelerometer, especially during periods of stationarity. In terms of the Wi-Fi sensor, extensive work exists for estimating the stability of the wireless signal environment [KKES10, KWRT12]. When a stable environment is observed, i.e., when the user is staying within a place, the Wi-Fi sensor can be switched off until notable kinematic movement is detected from the accelerometer [KKES10] or the primary GSM cell changes [RMB+10]. As humans tend to spend most of their time within a limited set of locations, with only occasional transition between these places [GHB08], significant reductions in power consumption could be achieved by minimizing the power consumption of the stationary classifier. Additional benefit from this line of strategy is that we could simultaneously monitor the user's significant places. In terms of accelerometer duty cycling, the accuracy of the gravity estimates is likely

to decrease as the sampling rate is decreased. As a part of our future work we plan to explore the robustness of these estimates when the accelerometer is duty cycled.

References

AY09 Atallah, L. and Yang, G.-Z., The use of pervasive sensing for behaviour profiling – a survey. *Pervasive and Mobile Computing*, 5,5(2009), pages 447–464.

BCMS01 Benini, L., Castelli, G., Macii, A. and Scarsi, R., Battery-driven dynamic power management. *IEEE Design & Test of Computers*, 18,2(2001), pages 53–60.

BDJN11 Bardram, J. E., Doryab, A., Jensen, R. M. and Nielsen, K. L., Phase recognition during surgical procedures using embedded and body-worn sensors. *International Conference on Pervasive Computing and Communications (PerCom)*, (2011), pages 45–53.

BI04 Bao, L. and Intille, S. S., Activity recognition from user-annotated acceleration data. *Proceedings of the 2nd International Conference on Pervasive Computing (PERVASIVE)*, Ferscha, A. and Mattern, F., editors, volume 3001 of *Lecture Notes in Computer Science*. Springer-Verlag, 2004, pages 1–17.

CAH08 Chen, C., Anton, S. and Helal, A., A brief survey of physical activity monitoring devices. Tech report mpcl-08-09, University of Florida, 2008.

CBC+08 Choudhury, T., Borriello, G., Consolvo, S., Haehnel, D., Harrison, B., Hemingway, B., Hightower, J., Klasnja, P. P., Koscher, K., LaMarca, A., Landay, J. A., LeGrand, L., Lester, J., Rahimi, A., Rea, A. and Wyatt, D., The mobile sensing platform: An embedded activity recognition system. *IEEE Pervasive Computing*, 7,2(2008), pages 32–41.

CKM+08 Consolvo, S., Klasnja, P., McDonald, D. W., Avrahami, D., Froehlich, J., LeGrand, L., Libby, R., Mosher, K. and Landay, J. A., Flowers or a robot army?: encouraging awareness & activity with personal, mobile displays. *Proceedings of the 10th International Conference on Ubiquitous Computing (Ubicomp)*. ACM, 2008, pages 54–63.

CMT+08 Consolvo, S., McDonald, D. W., Toscos, T., Chen, M. Y., Froehlich, J., Harrison, B., Klasnja, P., LaMarca, A., LeGrand, L., Libby, R., Smith, I. and Landay, J. A., Activity sensing in the wild: a field trial of ubifit garden. *CHI '08: Proceeding of the twenty-sixth annual SIGCHI*

conference on Human factors in computing systems, New York, NY, USA, 2008, ACM, pages 1797–1806.

CRTL04 Chan, C., Ryan, D. A. and Tudor-Locke, C., Health benefits of a pedometer-based physical activity intervention in sedentary workers. *Preventive Medicine*, 39(6)(2004), pages 1215–1222.

Die00 Dietterich, T. G., An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 40(2)(2000), pages 139–157.

FDFC10 Figo, D., Diniz, P., Ferreira, D. and Cardoso, J., Preprocessing techniques for context recognition from accelerometer data. *Personal and Ubiquitous Computing*, 14-7(2010), pages 645–662.

FDK+09 Froehlich, J., Dillahunt, T., Klasnja, P., Mankoff, J., Consolvo, S., Harrison, B. and Landay, J. A., Ubigreen: investigating a mobile tool for tracking and supporting green transportation habits. *Proceedings of the 27th International Conference on Human Factors in Computing Systems (CHI)*. ACM, 2009, pages 1043–1052.

FHT00 Friedman, J., Hastie, T. and Tibshirani, R., Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, 28,2(2000), pages 337 – 407.

FMT+99 Farringdon, J., Moore, A., Tilbury, N., Church, J. and Biemond, P., Wearable sensor badge and sensor jacket for context awareness. *Proceedings of the 3rd International Symposium on Wearable Computers (ISWC)*. IEEE, 1999, pages 107 –113.

FS95 Freund, Y. and Schapire, R. E., A decision-theoretic generalization of on-line learning and an application to boosting. *Proceedings of the Second European Conference on Computational Learning Theory*, 1995, pages 119–139.

FS99a Flinn, J. and Satyanarayanan, M., Energy-aware adaptation for mobile applications. *Proceedings of the seventeenth ACM Symposium on Operating Systems Principles (SOSP)*. ACM, 1999, pages 48–63.

FS99b Freund, Y. and Schapire, R. E., A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence*, 14(5)(1999), pages 771–780.

FS04 Flinn, J. and Satyanarayanan, M., Managing battery lifetime with energy-aware adaptation. *ACM Transactions on Computer Systems*, 22,2(2004), pages 137–179.

GCMB12 Gordon, D., Czerny, J., Miyaki, T. and Beigl, M., Energy-efficient activity recognition using prediction. *Proceedings of the Sixteenth International Symposium on Wearable Computers (ISWC 2012)*, (2012), pages 29–36.

GHB08 González, M. C., Hidalgo, C. A. and Barabási, A.-L., Understanding individual human mobility patterns. *Nature*, 453(2008), pages 779–782.

ICG05 Ichikawa, F., Chipchase, J. and Grignani, R., Where's the phone? a study of mobile phone location in public spaces. *Proceedings of the 2nd International Conference on Mobile Technology, Applications and Systems*. IEEE, 2005, pages 1 – 8.

JH98 Jaakkola, T. and Haussler, D., Exploiting generative models in discriminative classifiers. *Advances in Neural Information Processing Systems 11 - NIPS Conference*, (1998), pages 487–493.

KBBN11 Kjærgaard, M. B., Bhattacharya, S., Blunck, H. and Nurmi, P., Energy-efficient trajectory tracking for mobile devices. *Proceedings of the 9th International Conference on Mobile Systems, Applications and Services (MobiSys)*, 2011, pages 307–320.

KH04 Krumm, J. and Horvitz, E., LOCADIO: Inferring motion and location from Wi-Fi signal strengths. *Proceedings of the 1st International Conference on Mobile and Ubiquitous Systems (Mobiquitous)*. IEEE, 2004, pages 4 – 14.

KK98 Kravets, R. and Krishnan, P., Power management techniques for mobile communication. *Proceedings of the 4th annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98)*. ACM, 1998, pages 157–168.

KKES10 Kim, D. H., Kim, Y., Estrin, D. and Srivastava, M. B., Sensloc: sensing everyday places and paths using less energy. *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems (SenSys)*. ACM, 2010, pages 43–56.

KLGT09 Kjærgaard, M. B., Langdal, J., Godsk, T. and Toftkjær, T., EnTracked: energy-efficient robust position tracking for mobile devices. *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services (MobiSys'09)*, 2009, pages 221–234.

KM08 Kjærgaard, M. B. and Munk, C. V., Hyperbolic location fingerprinting: A calibration-free solution for handling differences in signal strength. *Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2008, pages 110–116.

KMS+08 Könönen, V., Mäntyjärvi, J., Similä, H., Pärkkä, J. and Ermes, M., A computationally light classification method for mobile wellness platforms. *Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine & Biology Society*, (2008), pages 1167–1170.

KSS03 Kern, N., Schiele, B. and Schmidt, A., Multi-sensor activity context detection for wearable computing. *Proceedings of the 1st European Symposium on Ambient Intelligence (EUSAI)*, Aarts, E. H. L., Collier, R., Van Loenen, E. and Boris de Ruyter, E. R., editors, volume 2875. Springer, 2003, pages 220–232.

KWRT12 Kjærgaard, M. B., Wirz, M., Rogger, D. and Tröster, G., Mobile sensing of pedestrian flocks in indoor environments using wifi signals. *Proceedings of the IEEE Pervasive Computing and Communication Conference (PerCom)*, 2012, pages 95–102.

LAA+09 Lazer, D., Adamic, A. P. L., Aral, S., Barabási, A.-L., Brewer, D., Christakis, N., Contractor, N., Fowler, J., Gutmann, M., Jebara, T., King, G., Macy, M., 2, D. R. and Alstyne, M. V., Computational social science. *Science*, 323,5915(2009), pages 721–723.

LCB06 Lester, J., Choudhury, T. and Borriello, G., A practical approach to recognizing physical activities. *Proceedings of the 4th International Conference on Pervasive Computing (PERVASIVE)*, volume 3968 of *Lecture*

Notes in Computer Science LNCS. Springer - Verlag, 2006, pages 1 – 16.

LCK+05 Lester, J., Choudhury, T., Kern, N., Borriello, G. and Hannaford, B., A hybrid discriminative/generative approach for modeling human activities. *In Proc. of the International Joint Conference on Artificial Intelligence (IJCAI*, 2005, pages 766–772.

LKD+10 Lau, S. L., Köning, I., David, K., Parandian, B., Carius-Dussel, C. and Schultz, M., Supporting patient monitoring using activity recognition with a smartphone. *7th International Symposium on Wireless Communication Systems (ISWCS*, (2010), pages 810–814.

LKLZ10 Lin, K., Kansal, A., Lymberopoulos, D. and Zhao, F., Energy-accuracy trade-off for continuous mobile device location. *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, (2010), pages 285–198.

LML+10 Lane, N. D., Miluzzo, E., Lu, H., Peebles, D., Choudhury, T. and Campbell, A. T., A survey of mobile phone sensing. *IEEE Communications Magazine*, 48,9(2010), pages 140 –150.

LWJ+04 Lukowicz, P., Ward, J. A., Junker, H., Stäger, M., Tröster, G., Atrash, A. and Starner, T., Recognizing workshop activity using body worn microphones and accelerometers. *Proceedings of 2nd International Confrence on Pervasive Computing (PERVASIVE)*, Ferscha, A. and Mattern, F., editors, volume 3001 of *Lecture Notes in Computer Science*. Springer, 2004, pages 18–32.

LXL+11 Lane, N. D., Xu, Y., Lu, H., Hu, S., Choudhury, T., Campbell, A. T. and Zhao, F., Enabling large-scale human activity inference on smartphones using community similarity networks (csn). *Proceedings of the 13th International Conference on Ubiquitous Computing (Ubicomp)*, 2011, pages 355–364.

LYL+10 Lu, H., Yang, J., Liu, Z., Lane, N. D., Choudhury, T. and Campell, A., The jigsaw continuous sensing engine for mobile phone applications. *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, 2010, pages 71–84.

MEBH08 Mun, M., Estrin, D., Burke, J. and Hansen, M., Parsimonious mobility classification using gsm and wifi traces. *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys)*, 2008, pages 1–5.

Miz05 Mizell, D., Using gravity to estimate accelerometer orientation. *Proc. Seventh IEEE International Symposium on Wearable Computers*, 18–21 Oct. 2005, pages 252–253.

MLMH07 Muthukrishnan, K., Lijding, M., Meratnia, N. and Havinga, P., Sensing Motion Using Spectral and Spatial Analysis of WLAN RSSI. *Proceedingds of the 2nd European Conference on Smart Sensing and Context (EuroSSC)*. Springer, 2007, pages 62–76,.

MPF+10 Musolesi, M., Piraccini, M., Fodor, K., Corradi, A. and Campbell, A. T., Supporting energy-efficient uploading strategies for continuous sensing applications on mobile phones. *Proceedings of the International Conference on Pervasive Computing (Pervasive)*, 2010, pages 355–372.

NBK10 Nurmi, P., Bhattacharya, S. and Kukkonen, J., A grid-based algorithm for on-device GSM positioning. *Proceedings of the 12th International Conference on Ubiquitous Computing (UbiComp)*, 2010, pages 227–236.

NJ02 Ng, A. and Jordan, M., On discrimative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in Neural Information Processing Systems*, 14(2002), pages 609–616.

PMI08 Papliatseyeu, A. and Mayora-Ibarra, O., Mobile habits: Inferring and predicting user activities with a location-aware smartphone. *International Symposium on Ubiquitous Computing and Ambient Intelligence*, (2008), pages 343–352.

PS01 Pillai, P. and Shin, K. G., Real-time dynamic voltage scaling for low-power embedded operating systems. *SIGOPS Operating Systems Review*, 35,5(2001), pages 89–102.

Qui86 Quinlan, J. R., Induction of decision trees. *Machine Learning*, 1(1)(1986), pages 81–106.

Qui93 Quinlan, J. R., *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

Rab89 Rabiner, L. R., A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2)(1989), pages 257–286.

RDML05 Ravi, N., Dandekar, N., Mysore, P. and Littman, M. L., Activity recognition from accelerometer data. *Proceedings of the 17th Innovative Applications of Artificial Intelligence Conference (IAAI)*, Veloso, M. M. and Kambhampati, S., editors. AAAI Press, 2005, pages 1541–1546.

RH97 Rubinstein, Y. D. and Hastie, T., Discriminative vs. informative learning. *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, AAAI Press, 1997, pages 49–53.

RMB+10 Reddy, S., Mun, M., Burke, J., Estrin, D., Hansen, M. and Srivastava, M., Using mobile phones to determine transportation modes. *ACM Transactions on Sensor Networks*, 6,2(2010), pages 13:1–13:27.

RSHI08 Ravi, N., Scott, J., Han, L. and Ifode, L., Context-aware battery management for mobile phones. *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2008, pages 224–233.

SC10 Shafer, I. and Chang, M., Movement detection for power-efficient smartphone wlan localization. *Proceedings of the 13th ACM International Conference on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWIM)*, 2010, pages 81–90.

SCB04 Schneider, P. L., Crouter, S. E. and Bassett, D. R., Pedometer measures of free-living physical activity: comparison of 13 models. *Medicine and Science in Sports and Exercise*, 36(2)(2004), pages 331–335.

SL91 Safavian, S. R. and Landgrebe, D., A survey of decision tree classifier methodology. *Transactions on Systems, Man and Cybernetics*, 21(3)(1991), pages 660–674.

SQBB10 Song, C., Qu, Z., Blumm, N. and Barabási, A.-L., Limits of predictability in human mobility. *Science*, 19,5968(2010), pages 1018–1021.

SVL+06 Sohn, T., Varshavsky, A., LaMarca, A., Chen, M. Y., Choudhury, T., Smith, I., Consolvo, S., Hightower, J., Griswold, W. G. and de Lara, E., Mobility detection using everyday GSM traces. *Proceedings of the 8th*

International Conference on Ubiquitous Computing (Ubicomp), 2006, pages 212–224.

TIL02 Tapia, E. M., Intille, S. S. and Larson, K., Activity recognition in the home using simple and ubiquitous sensors. *Proceedings of the 2nd International Conference on Pervasive Computing (PERVASIVE)*, Ferscha, A. and Mattern, F., editors, volume 3001 of *Lecture Notes in Computer Science*. Springer-Verlag, 2002, pages 158–175.

TLBJ04 Tudor-Locke, C. and Bassett Jr, D. R., How many steps/day are enough?: Preliminary pedometer indices for public health. *Sports Medicine*, 34(1)(2004), pages 1–8.

UBC97 Utgoff, P. E., Berkman, N. C. and Clouse, J. A., Decision tree induction based on efficient tree restructuring. *Machine Learning*, 29(1)(1997), pages 5–44.

WCM10 Wang, S., Chen, C. and Ma, J., Accelerometer based transportation mode recognition on mobile phones. *Asia-Pacific Conference on Wearable Computing Systems*, (2010), pages 44–46.

WLA+09 Wang, Y., Lin, J., Annavaram, M., Jacobson, Q. A., Hong, J., Krishnamachari, B. and Sadeh, N., A framework of energy efficient mobile sensing for automatic user state recognition. *Proceedings of the 7th International Conference on Mobile systems, Applications, and Services (MobiSys'09)*, 2009, pages 179–192.

WLG11 Ward, J. A., Lukowicz, P. and Gellersen, H. W., Performance metrics for activity recognition. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2,1(2011), pages 6:1–6:23.

WLT06 Ward, J., Lukowicz, P. and Tröster, G., Evaluating performance in continuous context recognition using event-driven error characterization. *Proceedings of the 2nd International Workshop on Location- and Context-Awareness (LoCA)*. Springer, 2006, pages 239–255.

WPP+07 Wang, S., Pentney, W., Popescu, A.-M., Choudhury, T. and Philipose, M., Common sense based joint training of human activity recognizers. *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, Veloso, M. M., editor, 2007, pages 2237–2242.

YAADH00 Yarlagadda, R., Ali, I., Al-Dhahir, N. and Hershey, J., GPS GDOP metric. *IEE Proceedings-Radar, Sonar Navigation*, volume 147, 2000, pages 259–264. Issue 5.

YSC+12 Yan, Z., Subbaraju, V., Chakrabarti, D., Misra, A. and K., A., Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach. *International Symposium on Wearable Computers (ISWC)*, (2012), pages 17–24.

ZLC+08 Zheng, Y., Li, Q., Chen, Y., Xie, X. and Ma, W.-Y., Understanding mobility based on gps data. *Proceedings of the 10th International Conference on Ubiquitous Computing (Ubicomp)*, 2008, pages 312–321.

ZLYX11 Zheng, Y., Liu, Y., Yuan, J. and Xie, X., Urban computing with taxicabs. *Proceedings of the 13th International Conference on Ubiquitous Computing (Ubicomp)*. ACM, 2011, pages 89–98.