Multi-Target Data Association and Identification in Binary Sensor Data

Von der Fakultät für Medizin und Gesundheitswissenschaften der Carl von Ossietzky Universität Oldenburg zur Erlangung des Grades und Titels eines

Doktor der Ingenieurwissenschaften (Dr.-Ing.)

angenommene Dissertation

von Herrn Sebastian Matthias Müller

geboren am 26.05.1983 in Nordenham

Erstgutachter:	Prof. Dr.-Ing. Andreas Hein
Zweitgutachter:	Prof. Dr.-Ing. Frank Wallhoff
Tag der Disputation:	20.05.2020

Statement of Authentication

I hereby declare that I have written the present thesis independently, without assistance from external parties and without use of other resources than those indicated. The ideas taken directly or indirectly from external sources (including electronic sources) are duly acknowledged in the text. The material, either in full or in part, has not been previously submitted for grading at this or any other academic institution.

Verfassererklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig, ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus fremden Quellen (einschließlich elektronischer Quellen) direkt oder indirekt übernommenen Gedanken sind ausnahmslos als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden.

Oldenburg, 10.10.2020

Sebastian M. Müller

Acknowledgements

First and foremost, I would like to thank my supervisor Prof. Dr.-Ing. Andreas Hein for giving me the chance to pursue this work. Thank you to the OFFIS Institute for Information Technology for supporting me in this endeavour from start to finish, believing in me and for giving me the opportunity to work on finishing this thesis full-time. Thanks also to Prof. Dr.-Ing. Frank Wallhoff for the very valuable feedback on such short notice.

This work also would not have been possible without my industrious colleagues and supervisors at OFFIS: Dr. Marco Eichelberg, Dr. Jochen Meyer, Dr. Wilfried Thoben, Andrea Fuhrken, Dr.-Ing. Frerk Müller-von Aschwege, Christian Lins and Alexander Gerka, to name a few. A special shout-out to my former colleague Patrick Schardin, the most proficient software developer I have ever met.

Special thanks to my parents, Rolf and Karin, without whom none of this would have been possible. I would like to thank my sister Melanie for being such a great role model to me all my life. Many thanks also to Manuel Gieseking, who patiently endured my incoherent ramblings about this work for too long. Finally, Anne, thank you for your patience, your kind words and for always being by my side.

Sebastian M. Müller

Summary

The growing availability of consumer hardware has made various use cases of activity monitoring possible. The use cases range from smart home automation to complex research activities, such as reproducing or assisting medical assessments. The fields of Ambient Assisted Living, Active Assisted Living, and Ambient Health Technologies have adopted many such devices to develop and provide services for people in need of care and supervision. Recent changes in demographic structure in Western society have caused a need for technical support systems for ambulant and residential care to enable people to live healthily and independently for as long as possible. The technology is used to improve resident safety and comfort in daily life, but also to actively and preemptively recognize possible health crises.

The more complex the activity or behavior that needs to be recognized – inactivity or absence is easier to detect than specific symptoms such as changes in gait or mental issues – the more data needs to be collected. The sensors required to do so are often perceived as an invasion of privacy and therefore commonly rejected in domestic settings. Conversely, low resolution sensors such as motion sensors and magnetic contact switches provide little data to derive relevant information from, but are not perceived as invasive and offer other benefits such as long battery life, being low-cost and easy to retrofit. Applications that require high-level data to measure or recognize complex activity patterns or the identity of a person must therefore find a compromise between resolution and reliability of the sensor data and practical concerns, such as users' perceived intrusion of privacy, form factor, price and power consumption. As a consequence, this work primarily aims to examine how much of the complex information required for activity monitoring can be recorded using low resolution ambient sensors, which require no cooperation or commitment of users.

The first task is to separate data from multiple persons moving in a space monitored by low resolution sensors. To do so, we present a modified multi-target tracking algorithm using Bayesian estimation and multi-hypothesis tracking. This algorithm allows for careful selection of data being stored based on confidence in the correctness of the data. It performs particularly well on low resolution data, such as when using cheap, off-the-shelf smart home sensors.

Multiple sensor events combined into complex activities or motion paths allow for elaborate activity analyses, but without identifying sensor data, activities can only be analyzed on a system-wide level, not per individual. Since many security- and care-related scenarios involve a person living with a partner or being cared for by visiting family, neighbors or care professionals, the data cannot differentiate between a person's data and that of a visitor and thus cannot provide information to support personalized medical or care-related decisions. The second part of this work thus focuses on the *long-term* separation of the activity data of residents and visitors. To avoid expensive and complicated setup procedures and lengthy periods of recording data for supervised learning approaches, we focus on clustering techniques to create individualized, pseudonymous motion profiles to separate residents.

As we will show, the information provided by low-resolution ambient sensors is often insuffi-

cient to implement a reliable multi-target tracking and identification solution. Many applications rectify this by making use of body-worn sensors. While wearing and maintaining – charging, syncing, replacing – such a sensor can be a burden to the user, some body-worn data is readily available through smartphones, smart watches, fitness trackers and other technology. If this data can be utilized for activity monitoring without causing additional demands, it can be used to improve both tracking and identification based on ambient sensor data. Therefore, we explore the extension of the ambient multi-target tracking algorithm by optional activity models derived from data of body-worn sensors.

We show that two residents of a two-bedroom apartment can be reliably tracked using basic ambient sensors only. We show that the tracking accuracy largely depends on the number and positioning of the sensors and that tracking is particularly difficult when two persons often cross paths or occupy the same space.

Combining ambient and body-worn sensors – such as acceleration sensor data from a smartphone – by improving the motion model of the tracking update function improves tracking accuracy. We show that the additional data helps tracking two residents while simultaneously providing additional *identifying* information. Most notably, accuracy was improved when *all* residents in our living lab study wore smartphones.

In the end, the requirements with regard to precision and accuracy towards a multi-target activity monitoring system heavily depends on the target application. To determine changes in gait speed over the span of several months such as during physical rehab, it is sufficient to identify and measure once a day while correct measurements and identification is indispensable. To enable personalized automation, such as light switching or heating, the resident must be permanently and reliably tracked and identified, though occasional malfunctioning rarely causes grave danger. For the former case, we have shown a way to solve the problem without additional hardware or complicated setup. For the latter case, we have implemented a way of incorporating data from body-worn sensors into the tracking algorithm.

Zusammenfassung

Die steigende Verfügbarkeit von günstiger und verlässlicher Verbraucher-Sensorik ermöglicht eine Vielzahl von neuartigen Anwendungsfällen des Aktivitätsmonitoring, von Smarthome-Automatisierung bis hin zu komplexen Forschungsvorhaben wie dem Ersetzen oder Unterstützen von klinischen Assessments. Forschungsfelder wie Ambient Assisted Living, Active Assisted Living oder Ambient Health Technologies zeigen, wie viele einfache, günstige und verfügbare Geräte eingesetzt werden können um neuartige Dienste für die häusliche (Pflege-)Unterstützung zu realisieren. Hintergrund für diese Arbeiten ist der steigende Bedarf an technischen Unterstützungssystemen, welcher durch die Änderungen in der demografischen Struktur westlicher Gesellschaften und den Wunsch vieler Menschen, möglichst lang ein unabhängiges Leben in den eigenen vier Wänden zu leben, bedingt ist. Der Einsatz von Technologie soll dabei helfen, Sicherheit und Komfort im täglichen Leben zu gewährleisten, aber auch mögliche gesundheitliche Schwierigkeiten frühzeitig zu erkennen.

Je komplexer die Aktivität bzw. das Verhalten ist, das erkannt oder vermessen werden soll, umso mehr Daten müssen erhoben werden. Die Sensoren, die dabei zum Einsatz kommen, werden häufig als Eingriff in die Privatsphäre wahrgenommen und deshalb von Endanwendern oft abgelehnt. Demgegenüber stehen niedrigauflösende Sensoren wie Bewegungsmelder und Kontaktsensoren, die zwar einzeln betrachtet wenig Informationen liefern, aber als weniger invasiv angesehen werden und zudem andere Vorteile wie einen einfachen Einbau, günstige Beschaffung und lange Akkulebensdauer mit sich bringen. Anwendungen, die komplexe Informationen erheben müssen, z.B. um Aktivitäten des täglichen Lebens zu erkennen und zu messen oder eine Person zu identifizieren, müssen deshalb einen Kompromiss zwischen Datenqualität und praktischen Belangen wie Nutzerakzeptanz, Preis, Formfaktor und Stromverbrauch finden. Deshalb ist die Hauptaufgabe dieser Arbeit, zu erörtern, inwieweit sich komplexe Aktivitätsdaten für häusliches Monitoring mit Hilfe von einfachen, niedrigauflösenden ambienten Sensoren, die keine Mitarbeit des Endanwenders verlangen, erheben lassen.

Zunächst müssen Sensordaten mehrerer Personen, die sich unter Umständen zur gleichen Zeit am gleichen Ort aufhalten, getrennt werden. Für diese Aufgabe präsentieren wir hier einen modifizierten Mehrpersonen-Tracking-Algorithmus basierend auf Bayesscher Schätzung und Mehrhypothesen-Tracking. Der Algorithmus ermöglicht es, bei unsicherer Datenlage Daten zu verwerfen, sodass nur möglichst präzise Daten tatsächlich erhoben und weiterverarbeitet werden. Der Algorithmus funktioniert besonders bei Daten mit geringer Auflösung, wie z.B. bei dem Einsatz von einfachen Smarthome-Sensoren, gut.

Die Verknüpfung mehrerer binärer Sensordaten zu komplexeren Aktivitäten oder Bewegungspfaden erlaubt eine ausführlichere Analyse von Aktivitäten. Ohne identifizierende Informationen kann diese Analyse aber nur systemweit, nicht auf individueller Ebene geschehen. Da viele sicherheits- und pflegerelevante Anwendungen die Erhebung von Daten in Mehrpersonen-Haushalten oder zumindest die Trennung von Daten von Bewohnern und Besuchern erfordern, können ohne Identifizierung keine Daten zur personalisierten Entscheidungsunterstützung er-

hoben werden. Der zweite Teil dieser Arbeit konzentriert sich deshalb auf die *langfristige* Trennung von Daten mehrerer Personen. Damit dies keine kostspieligen zusätzlichen Anforderungen wie eine komplizierte Einrichtung, zusätzliche Sensorik oder langwierige Datenerhebung stellt, konzentrieren wir uns hier auf Clustering-Methoden, die die bereits aufgezeichneten Sensordaten und die aus dem Mehrpersonen-Tracking resultierenden Aktivitätsinformationen nutzen, um pseudonyme Bewegungsprofile zu erstellen.

Diese Arbeit zeigt, dass zwei Personen in einer Wohnung mit zwei Schlafzimmern mit Hilfe von einfachen ambienten Sensoren verlässlich über einen langen Zeitraum nachverfolgt werden können. Sie zeigt ebenfalls, dass die Genauigkeit des Tracking-Algorithmus stark von der Anzahl und Positionierung der Sensoren abhängt und dass die Tracking-Genauigkeit gering ist, wenn zwei oder mehr Personen sich häufig begegnen bzw. den gleichen Raum einnehmen.

Um den Schwierigkeiten bei Tracking und Identifikation bei überlappenden Bewegungen/Aktivitäten zu begegnen, werden in solchen Fällen häufig hochauflösende oder körpernahe Sensoren hinzugenommen. Körpernahe Sensoren stellen für den Anwender aufgrund der erforderlichen Pflege und Wartung – Laden, Synchronisieren, An-/Ablegen – häufig eine zusätzliche Belastung dar. Oft sind solche Sensoren aber bereits im Einsatz, z.B. in Form von Smartphones, Smart-Watches, Fitness-Trackern und anderen Endbenutzer-Geräten. Wenn die Daten dieser Geräte also ohne zusätzlichen Aufwand für den Benutzer für Aktivitätsmonitoring genutzt werden können, können damit Tracking und Identifikation verbessert werden.

Diese Arbeit zeigt, dass die Kombination von ambienten und körpernahen Sensoren, z.B. der Beschleunigungssensor eines Smartphones, durch die Präzisierung des Bewegungsmodells in der Bewertungsfunktion des Tracking-Algorithmus seine Genauigkeit verbessert. In unserer Evaluation verbessern die Daten körpernaher Sensoren das Tracking von zwei Personen, während sie gleichzeitig Informationen zur *Identifikation* liefern. Der Effekt ist besonders bemerkbar, wenn *beide* Anwesenden körpernahe Sensoren tragen.

Abschließend lässt sich festhalten, dass die Anforderungen an ein Mehrpersonen-Tracking- und Aktivitäts-Monitoring-System stark vom angestrebten Anwendungszweck abhängt: Um Änderungen in der Gehgeschwindigkeit eines Reha-Patienten festzustellen, reicht es, die zurückgelegte Strecke der Person einmal täglich zu vermessen, während die korrekte Messung und Identifikation unerlässlich ist. Um eine personalisierte Smarthome-Automatisierung zu ermöglichen, muss die Person ständig verfolgt und identifiziert werden, eine gelegentliche Fehlmessung oder -identifikation bedeutet aber selten eine Gefahr. Für Anwendungen in ersterem Bereich zeigen wir einen erfolgreichen Ansatz ohne zusätzliche Hardware oder aufwendige Einrichtungsprozedur. Für Anwendungen in letzterem Bereich zeigt diese Arbeit einen Ansatz zur Integration von Daten körpernaher Sensoren in das auf Daten ambienter Sensoren basierende Tracking-Verfahren.

Contents

List of Figures

List of Tables

1 Introduction

The growing availability of cheap, reliable sensors and the desire to create technical support systems for a range of applications such as home automation and ambulant care have caused a surge of domestic activity monitoring applications. Such applications range from basic home automation tasks such as light and heating control to complex measurements such as medical assessments. The type of hardware used for these applications heavily depends on the type of data required and the circumstances of their usage. While some applications require high temporal and spatial resolution such as video cameras, the installation and use of such sensors in domestic environments is often objected to due to privacy concerns. Conversely, low resolution sensors such as magnetic contact switches provide little information but are less of a concern when used in private settings. Applications that require high-level data such as complex activity patterns or the identity of a person must therefore find a compromise between resolution and reliability of the data and users' perceived intrusion of privacy.

Some applications of domestic activity monitoring have gained popularity under the term *smart home automation*. Thereby, the term "smart" usually refers to simple sensor–actuator combinations such as switching lights when a motion sensor detects a person entering a room, adjusting the target temperature of a smart thermostat based on the time of day or triggering an alarm when a contact switch reports an open window or a smoke detector triggers. More complex automations are possible, but due to noise in sensor data and the complexity and variability of circumstances to detect, they often cannot rely on rule-based sensor–actuator bindings. Instead, they require the recognition of more complex behavior patterns including multiple sensor events in more or less strict order. This is a non-trivial task, especially when faced with noisy sensor data and when more than one person might be present at a time.

The field of Ambient Assisted Living (AAL) has adopted many smart home technologies to develop and provide services for people in need of ambulant care. Recent changes in demographic structure in Western society have caused an increase in the efforts to research technical support systems for ambulant and residential care and to enable people to live independently as long as possible. The technology is used to improve resident safety in daily life, but also to preemptively recognize possible health crises such as declining mobility or emergencies such as falls. Moreover, research has shown that clinical assessments, usually conducted by a health care professional in a clinical setting, can be partially replaced with measurements by ambient sensors in domestic environments. Instead of performing assessments in a clinical setting every few weeks, a monitoring system can record measurements of mobility and activity and automatically evaluate them without human intervention. Such measurements may include

time spent in certain locations of the house (bed, bathroom), gait speed, general mobility level and prolonged time frames of inactivity.

To be able to discover complex activities and behavior, many approaches involve the installation of high resolution sensors such as microphones, cameras or laser scanners. While these devices provide sufficient data to accomplish many tasks, including facial recognition, fall detection or posture analysis, they are often perceived as an invasion of privacy and therefore commonly rejected in domestic settings. Conversely, low resolution sensors such as motion sensors and magnetic contact switches are not perceived as invasive, but provide little data to derive relevant information from. As a consequence, this work aims to examine how much of the complex information required for activity monitoring can be recorded using low resolution sensors.

The primary task of this work is to determine to what extent multiple sensor events can be associated to one or more people without any (further) identifying or localizing information. Only then can more complex activity analysis happen. To do so requires at least a minimum of information on the spatial arrangement of sensors and the possibility for a person to move between them. Previous work has shown that such information can be derived from prerecorded data of the sensors alone [38].

To enable separation of low resolution sensor data from multiple persons, we present a modified multi-target tracking algorithm using Bayesian estimation and multi-hypothesis tracking. The algorithm makes no assumption on the type of technology or placement of sensors other than reporting any kind of motion or activity, but allows for careful selection of data being stored based on confidence in the correctness of the data. This approach is motivated by, but not limited to, applications of AAL. Instead of implementing specific services for a limited target audience, we develop approaches for tracking and separation based on minimal assumptions about the available data and the requirements of the application area.

The tracking space is defined by a graph, whereby the sensor areas are the nodes and their spatial adjacency the edges. Thus, it allows determining when there is more than one person present and helps to separate their activity. The algorithm should be designed to work on low resolution data first, such as when using basic smart home sensors, but should allow for the inclusion of more complex sensors and data as well. To study its precision and usability under varying conditions, we will test the algorithm across multiple setups varying by placement and number.

While ambient sensors are a reliable and convenient data source, many applications make use of body-worn sensors. While wearing and maintaining such a sensor – charging, syncing, replacing – can be a burden to the user, some body-worn data is readily available through smartphones, smart watches, fitness trackers and other technology. If this data can be utilized without causing additional costs, it can be used to improve both tracking and identification based on ambient sensor data. Therefore, we will also explore the extension of the above-mentioned multi-target tracking algorithm by *optional* activity models derived from data of body-worn sensors.

Multiple sensor events combined into complex activities or motion paths allow for more elab-

orate and precise activity analyses such as activity recognition. However, without identifying sensor data, activities can only be analyzed on a system-wide level, not per individual. Since many AAL scenarios involve a patient living with a partner or being cared for by visiting care professionals, the data cannot differentiate between a patient's data and that of a visitor and thus cannot provide information to support medical or care-related decisions. The second part of this work thus focuses on the long-term separation and identification of persons based on their activity data.

Most identifying sensors either require high-resolution data, such as video cameras, iris and fingerprint sensors, or they rely on discrete sensor readings that require specialized hardware as well as constant cooperation of residents, such as radio-frequency identification (RFID) through a body-worn RFID chip. Without identifying sensors, identification can only happen based on differences in behavior shown in the activity data. The task is therefore examined based on three approaches: first, separating activity data based on differences in behavior without any identifying information (i.e. *indexation*); second, probabilistic identification based on assumptions about differences in behavior and third, using identifying, body-worn sensors.

While numerous works have shown that identification of multiple targets in low resolution sensor data is possible [78, 96, 17], this work aims to show if and how this task can be performed under the additional constraint that information used to support medical and care related decisions are subject to more rigorous precision requirements than data used for smart home automation. At the same time, assistive technologies should be accessible and affordable, so they should not impose large financial costs or time and effort.

To accommodate all applications that require either separation *or* identification, we present an unsupervised learning approach to separating residents' data based on differences in activity data. This approach uses a fuzzy clustering algorithm that integrates constraints between data points based on temporal overlap. We evaluate various constraint-based clustering algorithms and show that the combination of fuzzy and constraint-based clustering performs better than each algorithm separately.

The approach can be extended to a semi-supervised solution by including identifying data into the process. This data usually comes from specific hardware, such as a RFID reader and chip, but the concept can be extended by probabilistic associations of sensors to identities or data from body-worn sensors. The goal of this approach is to enable various "degrees" of separability and identifiability based on the application's requirements and available data.

The remainder of this work is structured as follows: Section 2 describes the motivation behind AAL applications and the background of the problems described above. In Section 3, we cover previous works on various topics relevant to this work, such as multi-target tracking, mobility monitoring and binary sensor networks. In Section 4, we describe the approaches undertaken to solve each of the problems described above. Section 5 goes into detail how the algorithms and their surrounding architecture were implemented. Section 6 describes how the work was evaluated, several parameters that affect the performance of each algorithm and general results. Section 7 discusses the results and what has and has not been achieved. Section 8 describes several ways how this work could further be improved and what further work can be

derived from this one.

2 Motivation and Problem Statement

The problems of tracking and identification of multiple targets in low resolution data can be found not just in domestic applications, but in many more ranging from video surveillance of public spaces to sports analytics to autonomous driving. Many times, both problems are conflated and approached as one. Most of the time, privacy is not as much an issue as in domestic applications.

Section 2.1 and 2.2 describe two applications of multi-target tracking and identification and their societal implications. Sections 2.3 and 2.4 describes the technical challenges of tracking and identification in low resolution data. Section 2.5 formulates a problem statement.

2.1. Demographic Changes and Assistive Technologies

Recent changes in demographic structure have caused an increase in the efforts to research technical support systems for ambulant and residential care. The atomization of households [30], coinciding with a prolonged life expectancy [90], has increased demand for outpatient, home-based care. According to the German Federal Statistical Office, the ratio between ambulant care personnel supply and demand will halve between 2009 and 2030 while the number of single households increases. The number of single households in Germany is projected to increase by 600% in relation to the population numbers by 2030 [72]. Figure 2.1 shows the decline of the average household size in the United States between 1961 and 2011. Furthermore, the increased life expectancy (cf. Figure 2.2a) and improvements in medical care are causing a rise in the proportion of population living with chronic diseases: The ratio of economically dependent older population and the economically independent population is projected to double by 2050 (Figure 2.2b). At the same time, hospitals and care institutions are pushing towards outpatient care for economic reasons. According to the Annual Survey of the American Hospital Association [3], the percentage of revenue from outpatient versus ambulant care at community hospitals increased from 28% to 46% between 1994 and 2014.

The potential of using smart home installations to build assistive technologies for ambulant care support around cheap, ubiquitous hardware has been the subject of many studies in the fields of Ambient or Active and Assisted Living (AAL), Ambient Health Technologies and Assistive Health Technologies (AHT). Herein, smart home sensors and actuators are used to enable senior residents to live independently in their own homes for as long as possible. The technology is used to improve resident safety in daily life, but also to preemptively recognize possible health crises such as declining mobility or emergencies such as falls.

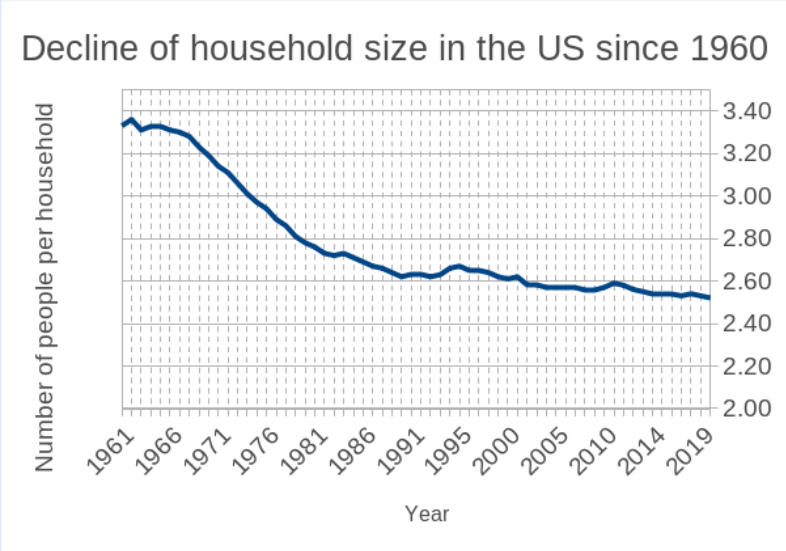
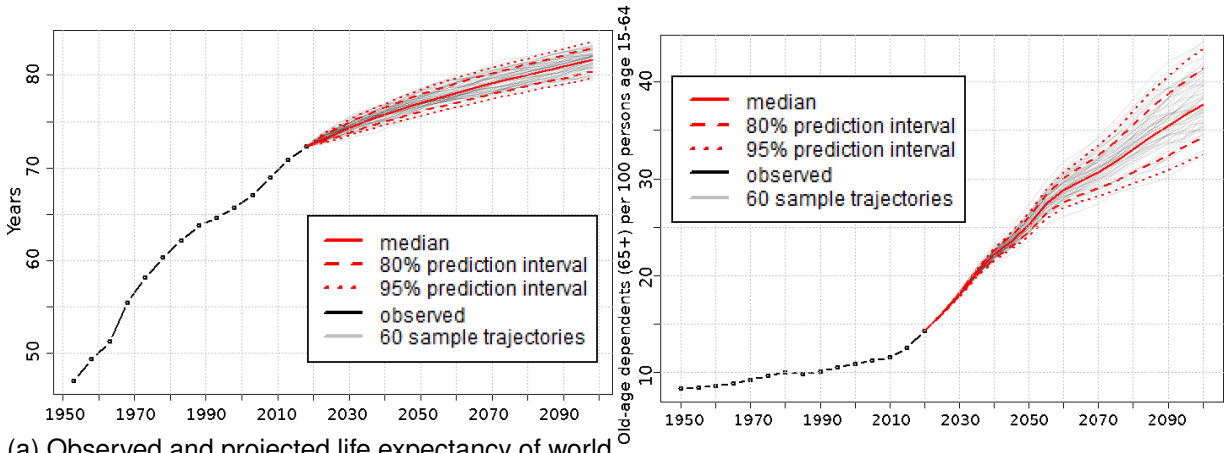


Figure 2.1.: Average household size in the US between 1961 and 2011. Source: U.S. Census Bureau [88]



(a) Observed and projected life expectancy of world population

(b) Ratio of dependents 65+ to persons of working age (per 100)

Figure 2.2.: Source: United Nations Department of Economic and Social Affairs [87]

In addition to smart home hardware, many modern medical devices are also equipped with networking technologies, making their data available for use in monitoring applications. For example, *Cicely*, a project funded by the German Federal Ministry of Education and Research (BMBF), combined medical sensors – blood pressure, blood oxygen sensors and scales – with ambient smart home sensors to collect data for palliative care support [37].

Many aspects of daily life that define a person's ability to live independently can be monitored, assessed and some even supported by assistive technologies. Ambient sensors (laser range scanners, passive-infrared motion sensors, contact switches) have been used in living environments not only help to conduct automated assessments, but they can support general care and daily living of inhabitants and enable *new* measures concerning mobility and activity through long-term ambient sensing. Instead of performing assessments in a clinical setting every few weeks, a monitoring system can record measurements of mobility and activity and automatically evaluate them immediately without human intervention. Such measurements may include time spent in certain locations of the house (bed, bathroom), gait speed, general mobility level and prolonged time frames of inactivity. These aspects can be categorized as *measures of mobility* and *activities of daily living*.

Mobility describes a person's ability to move freely and to assume and hold bodily postures. Since changes in mobility – beside aging – strongly correlate with changes in health [8, 14], it is of great interest to medical and care professionals. Thus, a battery of tests has been established that aims to measure changes in mobility of an individual over time. These tests are usually conducted by a health care professional in a clinical setting. For example, the Timed-Up-And-Go test consists of an individual performing a set of tasks involving getting up from a chair, walking a 3 meter distance, then turning around and sitting down again. Meanwhile, a health care professional observes the test and measures the time it takes the individual to perform the tasks from start to finish.

Due to the increasing pervasiveness and availability of affordable hardware, many attempts at automating health and mobility assessments are being undertaken. Main goal of these projects is to relieve the patients and health care professionals of performing the assessment, thereby simplifying the measurement procedure and reducing costs. However, a technologically assisted assessment also has the added benefit of measuring an individual's mobility *in-situ*, i.e. in their own home where they spend the majority of their time. This serves to create a more realistic measurement and removes the impact the test environment may have when the test is performed in a clinical setting.

Activities of daily living (ADLs) describe repeating activities of self-care and are often used to measure a person's functional status. These activities are often separated into basic and instrumental ADLs, where basic ADLs include walking, dressing, bathing, toileting, grooming and eating. Instrumental ADLs include activities which are required for independent living in the community, such as housework, shopping, transportation, managing finances and using the telephone [95]. The first use of ADLs as a geriatric assessment instrument was published in 1965 [56]. Since then, a number of assessment scales involving ADLs have been published, ranging from general geriatric [48] to Alzheimer's [36] to developmental disability assessments

[55]. Various studies have shown that several of these activities can be monitored, recognized and assessed using smart home sensors: Marschollek et al. [59] use ambient and wearable sensors in order to assess food preparation, personal hygiene and mobility. Tapia et al. [85] show that a wide range of ADLs (eating, bathing, grooming, dressing, meal preparation, etc.) can be recognized using vibration sensors and magnetic contact switches. Most studies, however, are limited to single-person households because activity recognition algorithms require largely noise-free data and two activities performed simultaneously are difficult to separate given the low resolution of the sensor data.

2.2. Home Automation

Home automation is the idea of automating processes in living environments in order to relieve inhabitants of recurring tasks, to improve safety and security of the inhabitants and to improve energy efficiency of a home. Recurring tasks may include lighting, sunscreening, heating, aeration, climate control and energy efficiency.

Ideally, home automation is fitted to the individual needs of residents. However, customization of services can be difficult after the installation process, where specific parameters such as target temperatures of automated heating control for day and night time are set. These existing solutions are therefore often considered inflexible and unsatisfactory. In order to further customize a home automation service, we must observe inhabitants during their daily lives and derive parameters from the data.

The most common smart home sensors are binary, meaning they signal only two states, such as passive-infrared (PIR) motion sensors or light barriers. Individually, they provide little – if any – information about personal behavior of residents. However, in large numbers they provide sufficient data to generate a more comprehensive activity and motion profile across all sensors and residents. Previous work has shown that binary sensors can be used to derive spatial structures of the home [38], to reconstruct motion paths of residents [97, 77] and to recognize activities [7]. Many such activities match those sought in care support, often as part of activities of daily living as part of an assessment instrument [51, 97].

While smart home automation revolves largely around convenience functions, it also has significant energy saving potential. According to the 2015 Residential Energy Consumption Survey [89], the average US household spends 42% of its total energy consumption on space heating, air conditioning and lighting (see Figure 2.3). Motion or presence detectors can turn light on and off based on activity. Programmable thermostats regulate room temperature according to a pre-scheduled program. For example, many users choose to lower the target temperature by a few degrees Celsius during the night. Additional smart home sensors can help develop a more personalized automation by registering presence and absence and environmental factors, such as outside temperature or solar irradiation. Harle and Hopper [40] have calculated that, in a 50-room office building, energy expended by lighting and “fast-response systems” such as computer monitors can be reduced by 50% with the help of an ultrasonic user localization system. Lu et al. [53] show that PIR motion sensors and magnetic contact switches

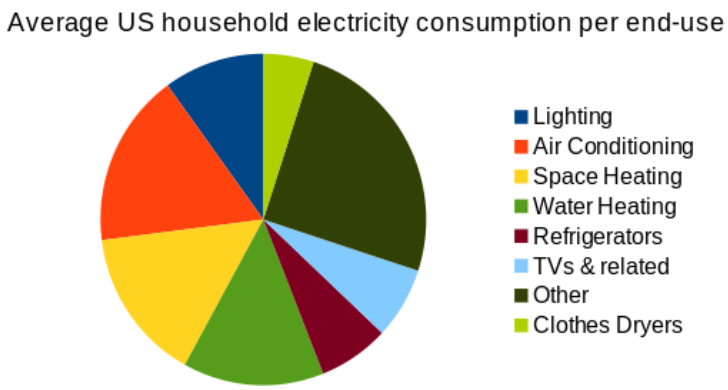


Figure 2.3.: Average annual household end-use expenditures in the U.S. Source: U.S. Energy Information Administration [89]

collect enough occupancy data to reduce HVAC (heating, ventilation and air conditioning) costs in residential homes by 28% “without sacrificing occupant comfort”.

Home automation hardware can be separated into radio-based and wired solutions. In the former case, sensors and actuators communicate via radio standards such as EnOcean, ZigBee or Z-Wave and are thus easier to install than wired hardware. In the latter case, common standards include BACnet, EEBus and KNX. The low-resolution sensors are also energy-efficient, meaning they can run on battery power for months or even years. Since the devices are running on battery and communicate wirelessly, they are easy to retrofit and rearrange in case the application requires it.

The resolution (both temporal and spatial) of a sensor and its power consumption often correlate: the requirement for long-lasting battery supply leads to low resolution, and low resolution allows for long-lasting battery supply. While applications such as lighting control based on presence do not require high temporal or spatial resolution, deriving higher-level information from the sensor data – motion profiles, identity, activities of daily living – is difficult. As such, these sensors also provide a sense of privacy compared to body-worn or higher-resolution ambient devices.

Protection of Privacy Many tasks being tackled by AAL and AHT solutions could be accomplished using high-resolution sensors. For example, video cameras provide enough data to perform complex tasks such as identification [100], recognition of activities of daily living [46] and fall detection [80]. However, cameras and microphones are often rejected for fear of loss of privacy [25, 97]. In fact, Morris et al. [62] conducted a literature survey of 1877 ““smart home” publications” and found that “[t]he primary barrier to the adoption of smart-home technologies by older adults was privacy concerns.” Conversely, motion sensors and home security systems are largely considered tolerable [97].

The degree by which a perceived loss of privacy is accepted in return for improved safety thereby strongly correlates with the users’ own perception of their health status [16]. However, the adequacy of the amount of personal data being recorded and stored is already mentioned

in laws such as the European Union's *General Data Protection Regulation* (GDPR) [28]. The principle of "data minimization" is the third of six principles relating to processing of personal data. The regulation states that "Personal data shall be [...] adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed" [28].

Body-worn devices such as RFID tags or accelerometers are often used to accomplish similar goals, such as localization [32] and mobility analysis [22]. Body-worn sensors offer *optional* privacy insofar as residents are free to take off the device when they feel observed or otherwise burdened. At the same time, these devices require active participation of the resident, who might take the device off and forget to or decide not to put it back on for reasons of (in)convenience. Body-worn sensors also require more maintenance because they are more restricted in terms of shape, size and power supply. As such, this work focuses on solutions based on ambient, rather than body-worn sensors.

In summary, low-resolution smart home sensors are easy to retrofit, comparatively cheap and require little maintenance while ensuring a sense of privacy. As such, they are ideal for domestic activity monitoring *as long as* they provide sufficient information for the application at hand. As was described in Section 2.1, many applications require separation or identification of multiple residents, which basic smart home sensors do not provide. The goal of this work, therefore, is to develop a multi-target tracking and identification system based on ambient smart home data that makes as little assumptions about the sensors, sensor data and other required information as possible.

2.3. Multi-Target Tracking

Multi-target tracking refers to the problem of jointly estimating the number and states – usually location and velocity – of targets from noisy sensor data. This problem relates to many applications in autonomous vehicles, robotics, air traffic control and others. As such, many solutions around a variety of sensors, including laser scanners, video cameras, RFID, infrared, ultrasound or Bluetooth badges have been developed. See Hightower and Borriello [42] for a survey of localization systems in ubiquitous computing.

The lower the temporal and spatial resolution of the data is, the less precise and reliable the estimated state of each target. Multi-target tracking with binary (smart home) sensors then deals with deriving target states under extreme uncertainty. Given the versatility, speed and precision of human motion, most of the motion data gets lost in the recordings of smart home sensors. While this appears to be a simple mismatch of hardware and application, this impreciseness is the root solving the privacy issue.

The very first step of tracking – separating targets – relies on data with sufficient resolution. If the location information is insufficient, a system may derive secondary information on the target, such as direction and velocity, by observing it over time. A common smart home motion sensor, however, may cover the whole area of a 4x4 meter room and merely report a binary signal. Therefore, we must find a way to optimize the multi-target tracking algorithm to find separation of targets with minimal distance *and* maintain correct association of sensor data to

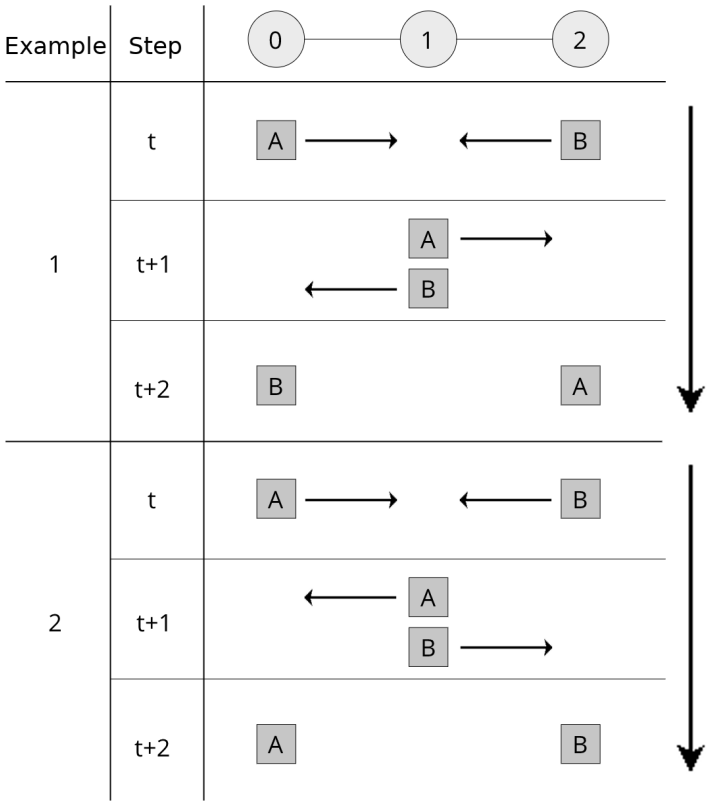


Figure 2.4.: Example of two scenarios difficult to differentiate based on binary sensor signals. The x-axis represents spatial movement, the y-axis time t

targets through situations of adjacent or overlapping activity.

Figure 2.4 shows two examples of motion paths of two targets across three binary motion sensors (0, 1 and 2). In the first example (upper half,), targets *A* and *B* switch places, crossing each other in the area of sensor *1*. In the second example, the targets meet in the same place, but then turn around and go back to their respective starting places. From the point of view of the three sensors, both scenarios look the same because there is no way to discern whether the targets turned around or kept walking.

In higher-resolution data, the tracking algorithm would make use of data beyond the locations of targets. For example, a flight radar can make certain assumptions about the speed, acceleration and maneuverability of a target, so that the possibility of observing the second example of Figure 2.4 can be excluded. To resolve, or better yet, avoid indistinguishable scenarios, we must either exclude them from further analysis in order to avoid introducing erroneous data into health- and security-related decision making, or we alter the data such that more relevant and/or more precise data is recorded.

2.4. Identification

Previous works have shown that a tracking algorithm can track an individual in a network of binary sensors with sufficient accuracy over short periods of time [24]. However, as Sullivan and Carlsson write,

> "Successful multi-target tracking requires solving two problems - localize the targets and label their identity. An isolated target's identity can be unambiguously preserved from one frame to the next. However, for long sequences of many moving targets, like a football game, grouping scenarios will occur in which identity labels cannot be maintained reliably by using continuity of motion or appearance." [83]

In their work, Sullivan and Carlsson use video data to generate spatially and temporally separated tracks for all players. Subsequently, they use information about the players' appearances and positions to derive likely connections between the tracks.

Unless additional (meta-)data or an identifying or body-worn sensor such as an RFID tag or biometric sensor is added, the resulting tracking data does not contain sufficient information in order to connect sensor events to create uninterrupted tracks or a motion or behavior profile and to detect individual changes in activity over time.

The second part of this work thus focuses on the use of binary sensors such as light barriers and motion sensors for the separation of residents for the *long-term* collection of activity data. Primarily, we want to investigate whether – while the amount of information provided by an individual binary sensor is insufficient – a network of low-resolution sensors collects sufficient data to differentiate between multiple residents over time.

If features of the activity data or metadata show differences between individuals, we can assume that tracks of binary signals provide sufficient information to separate two or more residents over large periods of time by grouping the data into "motion profiles": sets of tracks describing residents' activity over several days or weeks. This would not *identify* a person, but instead of having to associate an identity with each activity, it would be sufficient to associate an identity with a motion profile that contains large amounts of activity data over time.

Finding unique activity features among targets allows us to index users. That means, unless the sensor data also contains identifying information, we can assign them a pseudonym, but not an identity. Depending on the application, this kind of separation is sufficient. For example, when generating personalized lighting behavior for a smart home, the true identity of a person is irrelevant as long as they are distinguishable from others. Furthermore, mis-identification results in different system behavior, but likely would not cause any harm. When collecting health- and care-related data, however, the true identity is indispensable and even small errors might erroneously influence medical decision making. Therefore, a solution that supports both separation and identification, where identification sensors and data can be added as required, would be ideal.

2.5. Problem Statement

Cheap, reliable sensors are enabling a range of domestic activity monitoring applications, some of which with significant societal implications. In particular, the collection of care-related data from medical devices as well as ambient activity sensors has been the focus or a large body of research. The applications aim to improve the safety of residents in daily life, to recognize possible health crises such as declining mobility and to generate data that would otherwise only be collectable in a clinical assessment.

The recognition of complex behavior patterns or activities requires a lot of data that many users are hesitant to disclose. While smartphones and activity trackers generate vast amounts of data for the interested consumer, health-related applications must take reliability, costs and other stresses and strains that the system might impose on a user into consideration. Tracking and identification of multiple residents then becomes a non-trivial task, which is why many applications are limited to single-person households or demand the user wear a body-worn sensor.

Based on these observations, we can derive three unmet needs:

1. the ability to *quickly and easily* associate data from networks of low-resolution sensors to multiple targets,
2. the ability to *unobtrusively* separate data of multiple targets over long periods of time and across activities,
3. identification of targets *with little or no extra burden* in the form of active participation or expensive, possibly body-worn hardware.

3
State of the Art

The problems of tracking and identification of multiple targets in low resolution data can be found not just in domestic applications, but in many more ranging from video surveillance of public spaces to sports analytics to autonomous driving. Many times, both problems are conflated and approached as one. Most of the time, perceptions of privacy are not incorporated as they should be in domestic applications.

Large bodies of work exist around the topics of (a) multi-target tracking, (b) identification and (c) automated health- and mobility- or activity-assessments using smart home technologies. However, few cover the use of sensor networks *for* identification or the deduction of health and mobility data in multi-person households. Therefore, this chapter first describes the tracking and identification problems separately and from the point of view of applications whose constraints and solution spaces overlap at least partially with those of domestic applications. Afterwards, we have a look at some works around health- and care-related assessments that employ tracking and/or identification techniques. Note that *localization* of persons in the home is not the aim of this work, although target state estimation on discrete Bayesian networks can be understood as a form of localization. Without further specification, tracking does not impose any conditions on the *precision* of the localization. Therefore, this chapter does not cover any related work aimed only at localization. Section 3.1 covers applications of multi-target tracking and proposed solutions in private and public spaces. Section 3.2 describes previous works on identification in domestic environments using ambient and body-worn sensors. Section 3.3 describes applications of domestic activity monitoring that employ multi-target tracking and/or identification solutions. Finally, Section 3.4 summarizes the points of criticism of the state of the art.

3.1. Multi-Target Tracking in Multi-Person Households

This section presents a selection of relevant works around the topic of multi-target tracking using networks of low-resolution sensors.

Most recently, Wang and Cook [93] developed the “sMRT algorithm”, a Bayesian estimation model of sensor adjacency and transition probabilities to track multiple residents. The authors highlight the fact that, unlike most other tracking algorithms, sMRT does not require previous knowledge of the sensor setup, such as a floor plan or sensor graph. To generate the sensor graph, however, the approach relies on previously recorded data and the assumption that adjacent sensors are commonly activated subsequently. While this approach works well in many

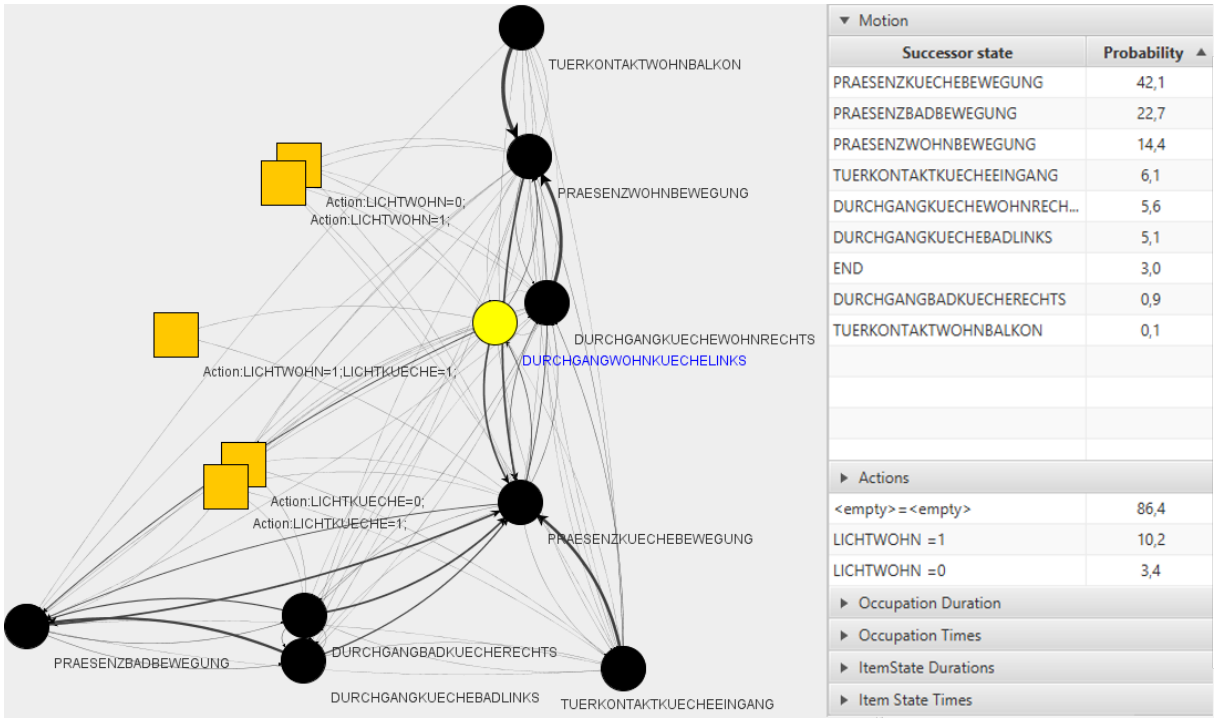


Figure 3.1.: Visualization of a personalized motion model with sensors (circles), actuators (rectangles), as well as motion and action probabilities (thicker arrows indicate more likely transitions; tabulated on the left) for the highlighted location (`DURCHGANGWOHNKUECHELINKS`)

cases, it performs poorly in noisy data: in the *Kyoto* dataset from the Washington State University, the dataset we use for our first evaluation, the algorithm accurately tracks two residents and occasional visitors 69% of the time.

In his PhD thesis, Daniel Wilson describes a system of ambient sensors and RFID readers to track and identify multiple residents using motion sensors, pressure mats and contact switches. Data association is implemented using a particle filter, where the state of each target is described by their room-level location and their activity state. The activity state takes the values `moving` and `not moving` and is derived from recent sensor events. Association probabilities are derived mainly from personalized motion models (i.e. motion or activity patterns expressed as, for example, state transition probability matrix or Hidden Markov Model) that are calculated from previously recorded data. Figure 3.1 shows the personalized motion model of a participant of the *LivingCare* project, recorded in a one-person household equipped with smart home sensors and actuators. The task is then to find the most likely sensor event associations based on target states and personal motion models. Wilson calls this the *Simultaneous Tracking & Activity Recognition problem*. [96]

The state of a single target can be estimated using recursive Bayesian filtering. Figure 3.2a depicts how the observed sensor data z is generated from a target's activity a and location r at time $t-1$ and t. To calculate the probability of a target's state x_t under the observed sensor data, we use Formula 3.1 (after Wilson [96]), where

- $p(z_t|X_t = x_t)$ is the likelihood of sensor data z_t being generated from target state x_t, referred to as the *sensor model*,
- $p(X_t = x_t|X_{t-1} = x\prime)$ is the likelihood of transitioning from state $x\prime$ to state x_t, also referred to as *motion model*, and
- $p(X_{t-1} = x\prime|z_{1:t-1})$ is the *a priori* probability.

$$p(X_t = x_t|z_{1:t}) \propto p(z_t|X_t = x_t) \sum_{x\prime \in X} p(X_t = x_t|X_{t-1} = x\prime)p(X_{t-1} = x\prime|z_{1:t-1}) \tag{3.1}$$

The state $x \in X$ at time t for each target m is $x_{mt} = \{r_{mt}, a_{mt}\}$, a combination of the target's location (r) and activity status (a), where a is derived from the most recent sensor events.

3.1.1. Sequential Monte Carlo (SMC) Method

As Wilson writes,

> "The [Bayes Filter] works well for tracking a single occupant in a noisy domain (the Bayes filter is named for its ability to filter spurious noise). However, this approach struggles to track multiple occupants because other occupants do not behave like noise processes. The tracker becomes confused by constantly conflicting sensor measurements. We need some way to determine which occupant generated what observation." [96]

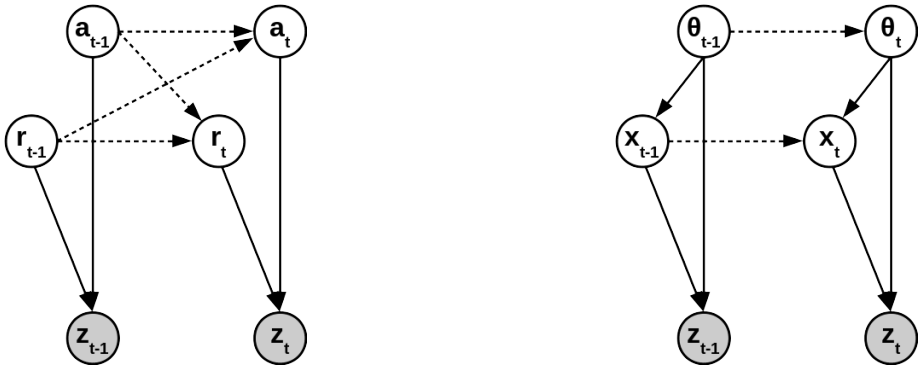
Other approaches, such as the Nearest Neighbor Standard Filter (NNSF) or Probability Data Association Filter (PDAF) fail for the same reason as the Bayes filter, namely that sensor events triggered by other people are treated as noise. Sequential Monte Carlo methods, also known as particle filtering, are currently the most prominent solution to multi-target tracking. Unlike Multi-Hypothesis Tracking (MHT) or Joint Probabilistic Data Association Filters (JPDAF), particle filters do not calculate every possible data association over time and is therefore computationally convenient.

Particle Filter This section describes the method as implemented by Wilson [96]. For a discussion of the performance of this approach, see Section 3.2.

To track multiple targets, Wilson incorporates the "sensor assignment matrix" (a matrix of associations of sensor events to targets) θ_t in the update formula. $\theta_t(i, j)$ is 1 if sensor event e_{it} was triggered by target j, 0 otherwise. Thus, the target states are calculated using the most likely association of sensor data to targets:

$$p(X_{1:t}, \theta_{1:t}|z_{1:t}) = p(X_{1:t}|\theta_{1:t}, z_{1:t})p(\theta_{1:t}|z_{1:t}) \tag{3.2}$$

A particle j at time t contains all targets' states $x_t^{(j)}$, a sensor assignment matrix $\theta_{1:t}^{(j)}$ and an "importance weight" $w_t^{(j)}$. The weight is derived from the posterior probability of the target states. The corresponding Bayesian network is depicted in Figure 3.2b.



(a) Discrete Bayesian network describing state estimation based on activity (a) and location (r)

(b) Discrete Bayesian network describing multiple targets' state estimation based on sensor assignments (θ) and individual target states (x)

Figure 3.2.: Discrete Bayesian networks as described by Wilson [96]

Particles are *sampled*, meaning the state space is considered too large to cover wholly. Instead, a number of particles is selected based on their importance weights. Then, new sensor assignments are generated and subsequently evaluated.

Other Works on Particle Filtering Oh and Sastry [66] describe an algorithm for distributed target tracking in a network of sensors. The problem is defined as a state estimation problem on a Hidden Markov Model (HMM). The problem is expanded to encompass the multi-target tracking problem as well as the possibility of having non-disjoint sensor areas. Unlike other approaches, this algorithm does not require a preceding calibration or training based on historical data. However, the sensor model does not account for noise, and the extension for multi-object tracking assumes that the sensors can identify one object from another.

Multiple studies describe the tracking of multiple targets based on Voronoi graphs [33, 50, 75]. A Voronoi graph consists of a number of points that are equidistant to at least two points on a line separating the space. If the "Voronoi areas" around two points touch, the points are connected by an edge [13]. This approach, thanks to data from non-binary sensors, has the advantage that localization of a person is not limited to a graph node (i.e. a single sensor). Schulz et al. [75] use a particle filter on data from laser scanners and infrared ID sensors, whereby each particle contains a Kalman filter describing associations of identity and location data of the targets. Background of the study is the idea that anonymous localization data is cheaper and more readily available than identification data. Therefore, a multi-hypothesis tracking algorithm (see Section 3.1.2) is employed to maintain tracks between identifying sensors. The location of targets is based on the data from laser scanners but is constrained to a Voronoi graph representation of the space. In an experiment with six participants recorded in an office setting with laser scanners, infrared and ultrasound receivers (to detect ID badges), they were able to retrace 29 out of 30 walking paths. In a different publication on the same work, Liao et al. attest that, despite the reduction of the state space, localization precision on a Voronoi graph is more precise that of an unrestricted particle filter [50].

Singh et al. [78] describe the *ClusterTrack* algorithm, a modified particle filter, to track multiple targets using proximity sensors. The modification prevents a single target from attracting all particles. Localization of targets happens not only on a sensor level: knowledge about the size of the sensor areas and distances between them helps locate the targets on a real-valued scale. This way, the evaluation function of the particles can make use of the estimated speed of a target, which is only reasonable if there are many sensors and their sensing area is small. The evaluation of the algorithm is limited to a simulation using targets with linear acceleration and an experiment using five proximity sensors with equal distances on a straight line.

Nguyen & Vekatesh [64] apply a particle filter on a stream of video images in order to detect "primitive and complex behaviors". To this end, the authors cover each image with a grid and associate motion paths through the grid with activities such as "cooking" and "eating a snack". The evaluation showed that the location of two persons was well recognizable even with short distances between them and that predefined activities were correctly recognized 79% of the time. It was also shown that the Joint Probabilistic Data Association Filter (JPDAF) was superior to a Kalman filter in associating activities to an individual.

As we can see, particle filters are the most common approach to multi-target tracking. They allow for flexible evaluation and resampling methods. This is particularly useful in large state spaces, such as in robot localization using laser range scanners or tracking of traffic in video and LIDAR[1] data for autonomous driving. In Wilson's work, the state space is large because targets are tracked over several days. Since many tracking applications do not require a target to be tracked over several days *and* smart home data is rarely high-resolution, we do not necessarily have to rely on probabilistic methods. One tracking algorithm that does not rely on sampling, but has not been thoroughly evaluated for use in domestic applications is *multi-hypothesis tracking*.

Wilson's work [96] has since its publication been extended to include Factorial Hidden Markov Model (FHMM)-based ADL recognition [2].

3.1.2. Multi-Hypothesis Tracking

The key principle of multi-hypothesis tracking (MHT) is to defer difficult data association decision to a later point in time when more data is available. The first complete description of MHT was given by Reid in 1979 [71]. The following is a general description of the algorithm after Samuel Blackman [10].

In multi-target MHT tracking, events are joined to *tracks*, such that a track represents the motion trajectory of a target. A hypothesis then consists of *compatible* tracks. Tracks are compatible if they have no events in common, meaning that a sensor event will only be associated with at most one target.

Each new sensor event can represent the start of a new track, an update to an existing track, or noise. So, from an existing hypothesis H_1 with two tracks T_1 and T_2 and a sensor event E_1 we derive four new hypotheses: one with a new track, two with an update to one of the existing

[1] light detection and ranging

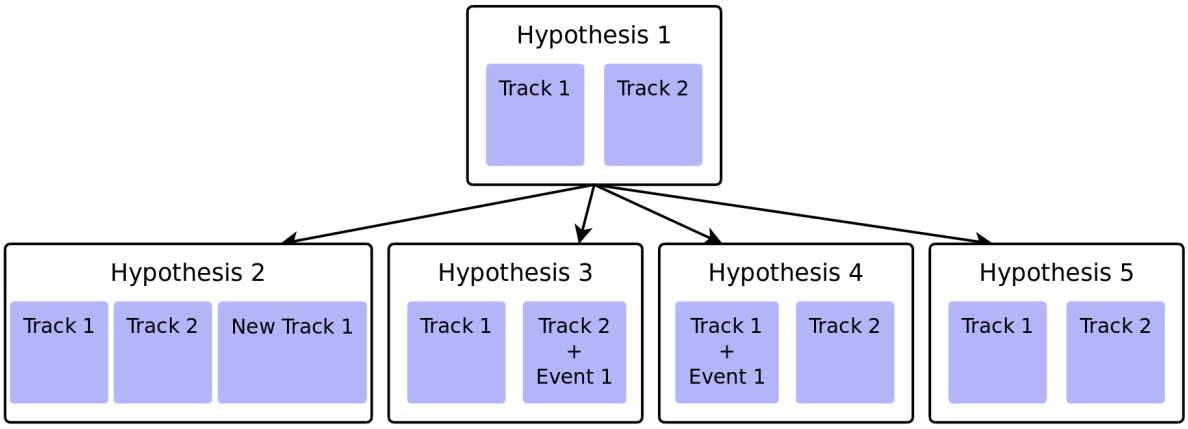


Figure 3.3.: Hypothesis generation from one hypothesis (H_1)

tracks and one where the event is discarded as noise:

- $H_2 : \{T_1, T_2, NT_1\}$
- $H_3 : \{T_1, (T_2, E_1)\}$
- $H_4 : \{(T_1, E_1), T_2\}$
- $H_5 : \{T_1, T_2\}$[2]

Figure 3.3 shows the hypothesis hierarchy visually.

Due to the large number of possible hypotheses that can be generated with large amounts of data and tracks, several techniques have been developed in order to optimize computation and not run into practical problems maintaining a large database of hypotheses. One method is clustering of tracks: if two tracks share at least one event, they belong to the same cluster. Once the tracks are separated into clusters, clusters can be processed in parallel. Another method is track pruning: tracks are represented as trees, where the nodes are sensor events and for every (possible) event, a new node and subtree is generated. The method is called N*-scan pruning*, where N stands for the maximum height of the tree. Blackman suggests "that N should generally be chosen to be at least 5". After every update, the root node is removed and the most likely track for each target becomes the new tree. The other tracks are pruned. [10]

Before new hypotheses are generated for a sensor event, a `gating` function filters all event-to-track associations in which the event is more than the maximum distance away from the last event in the track. The maximum distance can be chosen freely but is usually a function of the reliability of the sensors and the size of their sensing area. In the case of tracking on a graph, the gate size is a function of the distance of two nodes and the weights of the edges in between. Subsequently, all hypotheses are filtered based on various measures such as noise ratio, similarity and confidence. This procedure repeats until a single hypothesis remains or the

[2]Note that H_5 is the same as H_1 and as such not a "new" hypothesis.

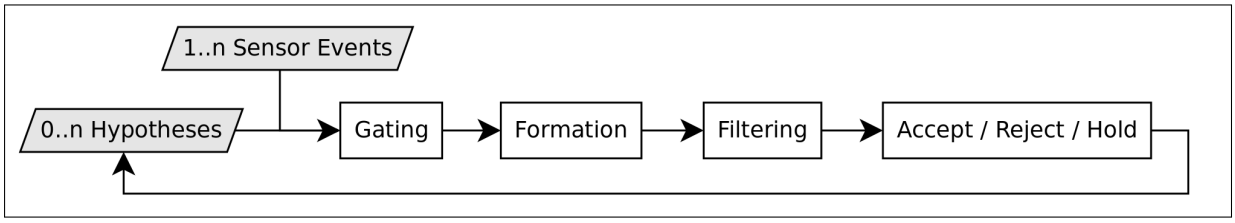


Figure 3.4.: Multi-hypothesis tracking procedure

window size is reached. In the former case, the hypothesis is accepted and the window size reset. In the latter case, all hypotheses are evaluated. If no dominating hypothesis is found, all hypotheses are discarded, the very first sensor event in the window and the underlying filters reset and the procedure repeated starting with the second event of the window. The high-level overview of this procedure is shown in Figure 3.4.

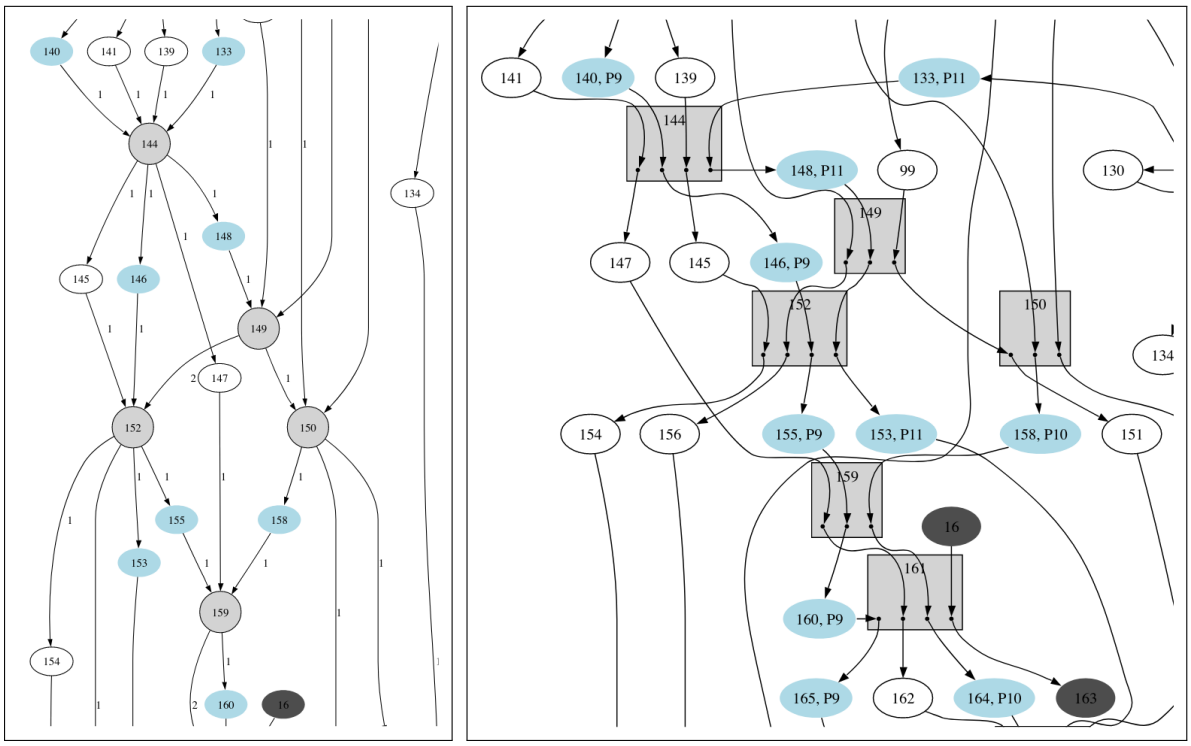
Multi-hypothesis tracking has been successfully used in various other applications, such as tracking targets in RADAR data [10], robot localization [5] and person tracking in laser scanner data [6].

3.2. Identification in Multi-Person Households

Previous work on separation and identification in ambient sensor data covers a wide range of technologies and algorithms. For example, Sullivan et al. [83, 84] generate single- and multi-target tracks from the stream of a video camera system of a football game to track players, then use unsupervised clustering to combine separate tracks of a player into one. Whenever two or more tracks merge due to lack of resolution or a track separates into multiple, the tracks' relationship is stored in the form of a graph. The result is what Sullivan et al. call a *track graph*, an example of which is depicted in Figure 3.5a. Next, the authors generate a similarity measure for the tracks based on the relative location of players to their teammates as well as temporal proximity. Starting with the longest single-target tracks, a cluster for each known target is initialized, then grown based on the greatest similarity of track to cluster and the fact that no tracks within a cluster can overlap temporally. Ideally, the result of this procedure is a resolved track graph as depicted in Figure 3.5b. The approach achieves 100% identification accuracy for tracks of 750 frames or longer, but would probably suffer under teams changing their formation mid-game due to the reliance on relative player positions [83].

In domestic environments, high-resolution sensors such as cameras and microphone arrays are commonly rejected for reasons of privacy. Wearable and body-worn sensors require continuous participation of the residents and are therefore often perceived as a burden. We will therefore limit this state of the art to approaches relying on low-resolution ambient sensors only.

As discussed in the previous section, Wilson [96] uses a particle filter to track multiple residents' location and activities in a home equipped with motion sensors, pressure mats and contact switches. Main goal of their work is room-level tracking and rudimentary activity recognition. Beside keeping track of presence count and motion paths, this approach also allows



(a) Example section of a track graph with nodes of unresolved tracks (gray)

(b) Resolved track graph

Figure 3.5.: Relationship of recognized track of football players depicted as a graph. Ovals represent single targets (team A (light blue), team B (white) and referee (black)), gray circles represent multi-target tracks. Source: Sullivan et al. [84]. Printed with permission.

identification of residents through singular RFID readers and motion models. In other words, the sensor events are not associated to an anonymous track (a string of sensor events), but to a resident.

In their evaluation, Wilson & Atkeson install an RFID lock at the front door, which identifies all residents as they enter the home. The identifying information is maintained over time by using personal motion models based on pre-recorded data for all residents. This requires recording the behavior of each resident and labeling their data. The motion model is defined by

$$p(X_t = x_t | X_{t-1} = x_{t-1}) = p(a_t | a_{t-1}, r_{t-1}) p(r_t | r_{t-1}, a_{t-1}) \tag{3.3}$$

where $p(r_t|r_{t-1}, a_{t-1})$ is the transition probability to a room given the previous room and activity status and $p(a_t|a_{t-1}, r_{t-1})$ probability of transition between activity states given the previous room and activity status.

The evaluation using simulation data shows that the tracking performance is highly dependent on these personalized motion models: the accuracy tracking two targets over one hour of simulated smart home data (motion sensors, contact switches, light barriers and pressure sensors) is 99% when the two motion models are maximally distinct and 46% when both targets use the same, uniform (i.e. transition probabilities are evenly distributed for all sensors) motion model. In another experiment, Wilson equipped a home with an RFID reader on the front door, 24 motion sensors (at least one in every room) and 24 contact switches (on doors, cabinets, drawers and refrigerator). To set up the system, the motion models are trained from five days of recorded and labeled data. With two residents, the tracker was 98% accurate. With three residents (each having their own study room and two sharing a bedroom), the tracker was still correct 84.6% of the time.

Similarly, Crandall [17] and Crandall and Cook [18] use the binary signal of PIR motion sensors, contact switches and other smart home sensors to train Markov models on labeled data of multiple residents. Based on this labeled data, the model is calculated based on the transition probabilities to S_t and the prior probability:

$$P(S_t|e_{1:t}) = P(e_t|S_t) \sum_s P(S_t|s_{t-1}) P(s_{t-1}|e_{1:t-1}) \tag{3.4}$$

To label an event sequence with a resident identifier, the authors use the most likely label based on the following formula, where i is the resident identifier and $e_{1:t}$ is a fixed-length sequence of events.

$$\arg\max_{i \in I} (i|e_{1:t}) = P(e_{1:t}|i) P(i) \tag{3.5}$$

In an experiment with two residents over five days in a living lab (Figure 3.6), this method was up to 92.4% accurate. In a similar experiment, the authors showed that a window size (the number of sensor events as input for the Markov model) of 25 was most accurate, while window sizes of 15 or fewer were well below 80% accurate. In a subsequent study, the same authors introduce BUG/ED, the “Bayesian Updating Graph based Entity Detector” [21]. This

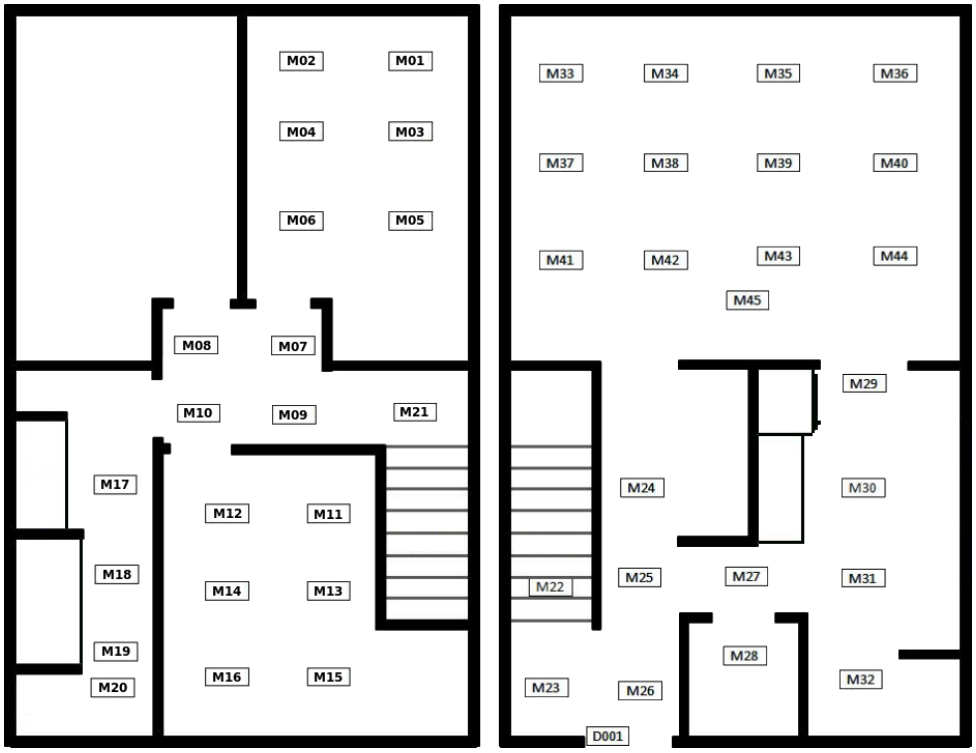


Figure 3.6.: Blueprint of the WSU CASAS living lab, showing the location of contact (D) and motion (M) sensors. After Crandall & Cook. [21]

approach does not require personal motion models, but instead uses a stream of sensor events annotated with the number of people present to derive a global transition probability matrix. On the data from the WSU CASAS *Kyoto* dataset, this approach manages to track approximately 60% correctly when two people are present.

Mokhtari et al. [61] built an ambient sensor system to identify residents of a smart home based on PIR motion sensors and an array of ultrasound sensors. The “Bluesound” system is installed in doorways and uses motion sensors to detect a person’s walking direction and an ultrasound sensor array to detect the person’s height. A linear discriminant analysis can distinguish between residents with an accuracy of at least 97.4% if the height difference is at least 4cm and as little as 59.5% if the height difference is 1cm.

Yun and Lee [98] developed a “data collection module” to detect direction, distance, speed and identity of a passing individual based on the raw data of three PIR sensor modules. Using three modules with four PIR sensors each, mounted in a hallway on two opposite walls and the ceiling, the authors achieve a 95% classification accuracy. Fang et al. [29] use a similar system – one module composed of up to eight PIR sensors with diverse detection regions – to identify up to 10 individuals using one trained Hidden Markov Model (HMM) for each individual and assigning the identity of the HMM with the highest likelihood of having produced the observation.

Zeng et al. [99] show that three residents can be distinguished based on gait parameters with the help of WiFi signal analysis with up to 90% accuracy. The work shows that the amplitude of

the Channel State Information (CSI) signal between a WiFi router and a WiFi-equipped device in the room resembles the signal from a body-worn accelerometer, thus enabling step detection. The "WiWho" system is trained to separate residents based on stepping and walking features derived from a previously recorded CSI signal.

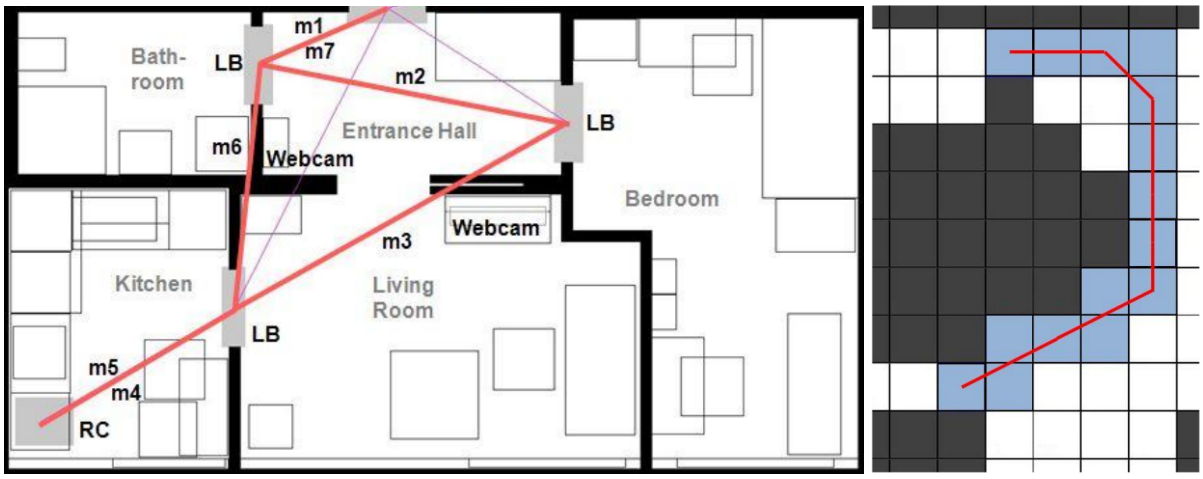
3.3. Mobility Monitoring and Assessment

Many of the works discussed in the previous sections mention mobility monitoring and other care-related data collections as a possible field of application. This section describes works that are specifically designed to solve issues in this field using low-resolution ambient sensors. Applications of such range from fall detection [67] to mobility evaluation and recognition of abnormal behavior [52, 41]. In conjunction with wireless medical sensors such as blood pressure and blood oxygen meters or scales, the data from these sensors helps assessing various aspects of a person's mobility.

Barger et al. [7] use mixture models to learn activity patterns of multiple individuals. Training data from motion sensors and contact switches are converted into complex activities in the form of a 4-tuple $(location, starttime, duration, activitylevel)$, then clustered using Expectation Maximization (EM). The evaluation is limited to a study with one individual. In the evaluation, the subject was asked to fill out an activity log, where all logged activities were labeled as one of 26 different activities. 66.3% of all clustered activities is found within one standard deviation from their respective cluster.

Steen et al. show that recordings from light barriers and reed contacts are sufficient to reproduce some clinical assessment tests [82]. While light barriers do not constitute sufficient evidence of a person entering a room (people may change directions between rooms), the authors suggest to combine them with sensors covering larger areas, such as ceiling-mounted motion sensors. Time and duration of habitation for all rooms are calculated by combining data from the sensors constituting a room. In a similar study, the authors derive a model of "normal behavior" of a person by calculating the location probability as well as the frequency of presence in a location by time of day and room (bathroom, bedroom, living room, kitchen) [81]. The models' performance strongly varies by room, as evidenced by the number of times anomalous behavior was correctly detected. In a similar study, Virone et al. [91] measure deviations of a patient's daily activities in terms of occupation times per room. The thresholds for anomaly detection are tuned manually.

Frenken et al. [34] use light barriers and reed contacts on doors to recognize motion patterns in order to conduct unobtrusive mobility assessments in domestic environments. The approach is not primarily designed for multi-person households, but it is similar to the tracking approach presented here insofar as the location of ambient sensors is used to generate a connectivity graph. In order to be able to generate the sensor graph, the authors rely on the provisioning of floor plans, in which the sensors and their sensing areas are located, then connected to a sensor graph using a path-finding algorithm. Figure 3.7b shows the structure of the path found between two light barriers, labeled *m6* in Figure 3.7a. Kitbutrawat et al. [44] show, however,



(a) Floor plan of a living lab equipped with light barriers and reed contacts. Calculated sensor graph depicted in red

(b) Calculated path for graph edge *m6* (left)

Figure 3.7.: Floor plan with sensor graph and corresponding calculated path. Source: Frenken et al. [34]. Printed with permission

that information on the proximity of two or more sensors can automatically be derived from previously recorded data. The advantage of using a floor plan is that the absolute distance between sensor areas can be determined, making mobility assessments more expressive. The disadvantage is, that floor plans are often hard to obtain and the digitalisation of the plan, its scale, the sensor locations and sensing area constitute a significant effort for each setup and household. In a different study [35], the same authors attempt to reproduce measurements of mobility and gait velocity as required in the Timed Up and Go assessment [68] using ambient sensors. In their evaluation across five apartments, the authors show that the data is sufficient to compute gait velocity – part of the Timed Up and Go – unobtrusively at home.

Ogawa et al. [65] collected data from homes instrumented with motion sensors and contact switches for one year in order to train models for classification of activities of daily living. Patterns of activity, including “absence”, “use of stove” and sleeping times were “clearly identifiable”. Barger et al. developed models to detect ADLs based on data from an array of motion detectors and contact switches [7]. Sensor readings are clustered to groups based on room, duration, and time of day to show that many clusters correspond to ADLs.

Logan et al. [51] tested motion sensors, RFID tags, on-body as well as on-object sensors for detection and classification of activities of daily living. The motion sensors yielded the best results overall, although it is pointed out that the classification performance was best on activities that are bound to specific locations in the home, such as “watching TV” and “meal preparation”.

Das and Cook [23] developed a model to identify and predict daily activities for the detection of short-term anomalies. An extended *Apriori* algorithm [1] detects activities based on recurring sequences of sensor events as well as long-term changes. A living lab study evaluated the recognition of recurring activity patterns: on data of six participants, spanning a total of one month, self-reported activity patterns were correctly detected in 47% of cases.

Shin et al. [76] employ motion sensors to generate normal behavior models for the detection of unusual behavior. Behavior is modeled as “activity level, mobility level and non-response interval (NRI)”. “Activity level” describes activity by summing up the number of motion sensor events caused during each time slot. “Mobility level” describes movement by counting location changes as evidenced by two different sensors triggering consecutively. The “non-response interval” describes time and duration of inactivity measured by the time between two movements. Using an approach coined “support vector data description”, normal behavior and anomalies are correctly classified with an accuracy of more than 90%.

Floeck et al. [31] use smart home sensors to identify anomalies in behavior by means of “inactivity profiles”, whereby inactivity is defined as the time between two sensor events. Goal is to classify a day as normal or abnormal by comparing an inactivity vector with a corresponding time slot in the reference vector defining normal behavior. Thereby, an inactivity value is equal to the corresponding reference value if it is within a predefined range of tolerance of the reference value. In an extended version of the approach, the maximum duration of inactivity over several days is calculated to distinguish normal and anomalous behavior. The threshold is defined by the maximum observed duration of inactivity plus 30 minutes. When abnormal behavior is detected, an alert is generated.

Skubic et al. [79] model normal behavior (bathroom activity, bed restlessness, kitchen activity, living room activity) based on data from motion, temperature and pneumatic bed sensors in order to generate alarms for detected anomalies. Models are computed for three time periods: a 24-hour day, daytime only (8am to 8pm) and nighttime only (midnight to 6am). In their evaluation, the model is based on data from a single-person household recorded over the span of two weeks. The models are trained using fuzzy pattern tree and support vector machine classification. Both classifiers achieved similar results in detecting anomalies.

Where ambient sensors are insufficient, body-worn and ambient sensors are fused: Wang et al. [94] merge ambient and wrist-worn sensors to detect and anticipate activity patterns. Chapron et al. [12] merge ambient and body-worn sensors to measure and assess gait speed and simultaneously associate the measurement with the right user. However, the former study also relies on supervised learning and the latter on identification based on personal body-worn sensors. As one of a few studies in this area, a recent publication by Hossain et al. [43] addresses the issue of requiring labeled data for model training, in this instance for activity recognition: by applying an *active learning* approach to the training of a deep neural network (“active deep learning”), a few labeled instances are sufficient to build an activity recognition model that is superior to comparable approaches in most cases. Active learning does not, however, fully eliminate the task of recording and labeling training data.

3.4. Critique of the State of the Art

The works presented in the previous section are only a small example of the work done around domestic mobility monitoring and AAL-related applications of ambient sensing, but they show a clear shortcoming: none of the works was evaluated in a multi-person household. While some

of these applications are certainly well-suited and even developed for single-person households, their adoption is clearly hindered by this deficit. Thereby, not all applications require definite identification at all times, but would be improved by even short term, anonymous tracking.

3.4.1. Multi-Target Tracking in Multi-Person Households

As we have seen, particle filters and similar probabilistic methods are the most common approach to multi-target tracking. As Wilson states,

> "For t seconds and m occupants each association has $m!t$ possibilities. In a reasonable scenario with several dozen inexpensive sensors monitoring a handful of occupants for a week, there are too many data assignments to enumerate." [96]

However, in order to collect data for personalized lighting automation or clinical assessments such as gait speed analysis, it is not necessary to track targets over several days. Most applications require only tracking across a few sensor events, based on the number, resolution and density of sensors. Rather than trying to collect vast amounts of sensor data over long time periods, then sampling possible solutions, we can reduce the length of observation and search the solution space exhaustively.

Furthermore, when dealing with health-related applications, we wish to be as certain as possible about the information produced, so the ability to discard uncertain data and reevaluate events in case of uncertainty is useful. While it is possible to employ evaluation and resampling methods that discard unlikely particles, other methods lend themselves more directly to the kind of short-term tracking that is required.

Lastly, no works exist that cover the *practical* requirements for a sensor network for activity tracking and assessments in multi-person households. Wilson [96] shows how their particle filter performs using varying numbers of particles and residents, but not using varying numbers of sensors. Singh et al. [78] address the theoretical countability of targets in binary sensor networks. However, an evaluation of their approach applied to multi-target tracking for mobility assessments was not provided.

3.4.2. Identification in Multi-Person Households

All identification approaches described in this chapter have one thing in common: they rely on labeled data to generate personalized motion models or train classifiers. Supervised learning brings certain advantages: the correctness of the system can be verified immediately after learning and the residents can not just be separated but also identified. However, the recording of training data is an elaborate task, which usually takes even a trained professional several hours, if not days. If these systems are to be used in hundreds or thousands of households, the labeling of said data will be a costly requirement.

Two notable related works ([96, 51] use RFID to associate activity with an identity, an approach that has two severe weaknesses: first, should the RFID chips be swapped for any

reason, all subsequent associations of activities and identities will be incorrect. Second, the failure of the single RFID reader would cause the loss of any and all identifying information.

3.4.3. Summary

In summary, we can deduce three points of critique:

1. Previous multi-person tracking approaches focus too much on tracking over long periods of time and too little on short term correctness and certainty to be useful in supporting health-related decision making.

2. Most multi-person tracking solutions based on low-resolution data rely on pre-recorded data and supervised learning to derive personalized motion models and/or metadata on sensor areas and the sensing space. This process is time-consuming and costly and hinders adoption.

3. Identification of persons in low-resolution data relies on either body-worn devices or pre-recorded data. Especially care-related applications might be time-critical, rendering data recording for days or weeks unacceptable. Furthermore, many care patients cannot be expected to remember – or even accept – to maintain and constantly wear a body-worn sensor.

4 Approach

In this chapter, we will outline the technical approach to solving the two key problems – multi-target tracking and identification – under the conditions imposed by the target application areas. We also delineate the differences of the target applications to similar applications and solutions and place the solution in two taxonomies found in literature. Finally, we will summarize the main contributions of this thesis.

4.1. Multi-Target Tracking

This thesis aims to show how data that may help assessing multiple persons' activity as well as a person's health and mobility can be recorded using simple smart home sensors. The main technical requirements for such a system are that it works without a complicated or expensive setup procedure and without burdening the users with permanently having to cooperate, such as carrying a body-worn sensor at all times. The main functional requirement towards this system includes the separation of activities from multiple residents as well as those of visitors and possibly pets.

Information that might help assessing a person's health and mobility are those that have already been covered in various research projects in the areas of ambient assisted living and tele-health, among others

- Activities of Daily Living (ADLs),
- Quantification of physical (in)activity,
- Measurements congruent with clinical mobility assessments or
- long term trends and changes in motion and activity patterns.

From a technical point of view, the focus of this work lies on the collection of data to support health- and care-related services and decision making using a system that requires little to no cooperation from the users. It also should not make assumptions or demands towards the type of hardware used so as to enable service providers and end users to reuse hardware used for different purposes, such as smart home services. The fact that simple ambient sensors can be more useful for recognizing activities than body-worn sensors has already been shown [51].

From a user's point of view, there are three essential benefits to this approach: first, requirements towards the hardware is minimal. Any sensing technology which correlates with

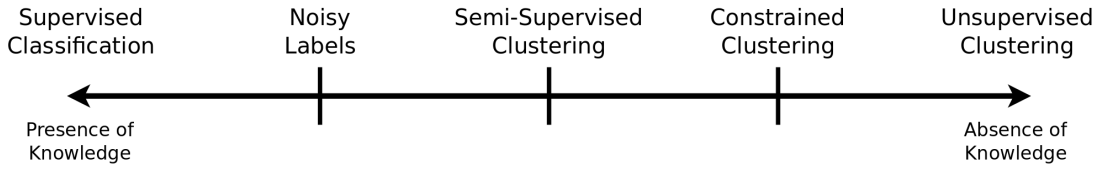


Figure 4.1.: Spectrum of clustering approaches based on availability of class membership data. After Bouchachia and Pedrycz [11]

presence or activity of a person, such as passive-infrared motion sensors, contact switches or motion-triggered cameras, can be used. Second, this approach requires no complex setup or installation procedure. Unlike other approaches, no historic or labeled data is required to learn motion or sensor models, no complex calibration procedure must be conducted after installation and no specialized hardware must be installed. Third, low-resolution sensors protect the privacy of the residents early on. While large amounts of activity data can be used to create a motion model of residents, which in turn can be used to derive personal habits, this can only be done using complex algorithms and prevents collection of highly personal data like cameras and microphones do. [112]

4.2. Identification

As Singh et al. [78], Wilson et al. [96] and others have shown, a tracking algorithm can track individuals over short periods of time based on data from networks of binary sensors with (for many applications) sufficient accuracy. However, without additional, identifying data such as from an RFID tag or biometric sensor, there is not sufficient information to track a person over several days or weeks or to “connect” tracks to create a motion or behavior profile. Only then would it be possible to detect individual changes in activity over time.

Figure 4.1 represents the whole problem space from supervised classification, where complete knowledge of the association of instances to classes exists, to unsupervised clustering, where no such knowledge exists. Cases in which *some* tracks are labeled due to the existence of a (perfectly) identifying sensor fall into the center of the line (*semi-supervised clustering*).

We hypothesize that tracks of binary signals of sufficient length provide enough information to cluster the tracks into “motion profiles”, sets of tracks describing a person’s regular motion and activity over several days or weeks. To test this hypothesis, we employ three clustering methods: fuzzy clustering, constrained clustering and constrained fuzzy clustering. As feature space, we identify data and metadata that can be used to distinguish tracks of all residents and guests, including activity in unique locations, at unique times of the day or relative gait speed. [113]

Since clustering based on binary ambient sensor data helps associating tracking data to pseudonyms but not identities – there is no identifying information in the sensor data, merely similarities – we then add different “levels” of identifying information. We start by adding associations of sensor events to identities based on previous knowledge of behavior in the form of clustering constraints. Finally, we emulate and add identifying sensors, similar to how an

RFID sensor or facial recognition system would work. At this stage, *active learning* approaches as used by Antoine et al. [4] or semi-supervised clustering as suggested by Bouchachia and Pedrycz [11] could be applied. However, for the sake of comparability and compatibility, we focus on the evaluation of the above-mentioned fuzzy and constrained clustering algorithms.

4.3. Fusion of Ambient and Body-Worn Sensors

As we will see, a purely ambient tracking system based on binary sensor data struggles to separate targets when they are close to another. This problem is commonly solved by generating personalized motion models using pre-recorded data and supervised learning techniques. However, this is a significant burden for users and a barrier for adoption. Therefore, we present an extension of the multi-target tracking algorithm that incorporates data from body-worn sensors to amend the motion model of the tracking update function. It is imperative that wearing such a sensor remains optional and does not impede the tracking functionality when none or only a subset of the targets carry such a sensor.

The motion model is also designed to be generic: while certain sensing technologies perform better in recognizing motion and activities – gyroscopes and accelerometers are common – the model interface is designed to be generic. The data source can be a step counter, heart rate monitor, smartphone or any other device that allows for timely delivery of activity data.

4.4. Solution Classification

To illustrate the differences between the application areas targeted in this work and other tracking and identification applications, we can classify them by various taxonomies proposed in literature surveys of the space.

In their “survey of human-sensing”, Teixeira et al. [86] identify five low-level components of the sensing of “human spatio-temporal properties”:

1. Presence (“Is at least one person present?”),
2. Count (“How many are present?”),
3. Location (“Where is this person?”),
4. Track (“How did this person move?”),
5. Identity (“Who is this person?”)

Each component presupposes all previous components: by determining *count*, we also answered the question of *presence*; by determining *location*, we also answered the question of *count*, and so on. While each component serves to answer a question which can be helpful in smart home and assisted living applications, and there exists a system optimized for answering individual components’ questions, the most ubiquitously useful information is drawn from

the last two components, tracking and identification. By focusing on these two, we hope to approach a solution answering all five questions.

The work's "summary of capabilities of each sensing modality" illustrates the tension between the available hardware and sensing goals. All common smart home sensing modalities (motion, pressure, electric field, vibration) are classified as having "low" or "no" detection performance with regard to *identity*. Motion and vibration sensors are also classified as "low performance" with regard to *count*, *location* and *track*. [86]

Savvides et al. [74] describe nine axes along which requirements of localization applications fall. This taxonomy does not cover all necessary requirements for our use cases. Most notably, the issue of privacy protection is not mentioned. However, even though localization is only a part of the solution, the requirements discussed fit domestic tracking and identification applications well.

Granularity and Scale of Measurements With no knowledge of the size and layout of the sensor space, tracking and localization happens on a graph representing relative positions. Depending on the sensor, a single graph node may cover less than a square meter (contact switch) or a 5x5 meter room. With at least one sensor per room, we characterize this as *at or below room-level granularity*.

Accuracy and Precision Mobility assessments require maximal accuracy and precision in order to avoid generating misleading data and to avoid miscounceling. Smart home applications have a slightly larger margin of error, though the user experience quickly suffers from a lack of reliability.

Relation to Established Coordinate System Relative positioning is sufficient for all target tasks. While many mobility assessments require measurements along absolute axes, such as the 3 meter walking distance during the Timed-Up-And-Go test, this would again require a blueprint of the sensor space or manual measurements, which we do not assume to have access to.

Dynamics Sensors are fixed in their location and orientation, though occasional outages may occur. For power saving reasons, many smart home sensors have a down time of several seconds after an activation. While this may harm accuracy and precision of tracking, the system must accommodate it.

Cost Monetary, power consumption and installation cost per sensor should be low and the system should be quick to install. An assistive system should be available for as many people as possible and it might need to be installed on short notice, such as after a fall or a break-in.

Form Factor The hardware should not interfere with the daily life of residents. Smart home hardware can usually be installed on ceilings, walls, doors, windows and appliances with-

out being a hindrance. The communication range is sufficient for most homes; for example, *ZigBee*[1] and *FS20*[2] have a theoretical range of up to 100m.

Communications Requirements No communication *between* sensors is necessary, but real-time reporting of sensor data is necessary for many smart home applications. Some assisted living and care applications may use communication to the outside, such as when reporting incidents.

Environment Indoor only.

Target Cooperation Users should not be required to actively participate in order to collect data. In some cases, the system can and should "act", such as for light automation or reminding a user of appointments or medications.

Having specified the system requirements, we can now describe the contributions of this work towards producing a system that matches the above-mentioned use-cases and requirements.

4.5. Thesis Contributions

This research explores the field of tracking and identification of residents for applications of smart home automation and, more importantly, health- and care-related decision support. This work describes the development of a combination of tracking and identification procedures that cover several important shortcomings of the current state of the art.

Adaptation of a multi-hypothesis multi-target tracking algorithms for binary sensor networks: We implement an adapted version of the multi-hypothesis tracking (MHT) algorithm that favors correctness over completeness. This approach facilitates the use of the resulting data for critical applications, such as medical decision support. It does not include identification, so that applications that require tracking (but not identification) can take advantage of it without requiring additional hardware or a period of recording and labeling training data for activity modeling.

A method for aggregation of anonymous activity data to pseudonymous activity profiles: We show how anonymous tracking data can be aggregated to pseudonymous motion profiles by taking advantage of inter-personal differences in activity data. This approach facilitates identification in applications where identifying data is sparse or simply not needed. We show how the addition of even uncertain identification data helps generating motion profiles and enables collection of long-term mobility data.

[1] https://zigbeealliance.org/solution/zigbee/
[2] https://www.eq-3.com/products/homematic.html

Integration of ambient and body-worn sensors for tracking and identification in binary sensor networks: We extend the multi-target tracking algorithm by an extended motion model to incorporate data from body-worn sensors. This model allows the inclusion of non-stationary data, such as from smartphones or fitness trackers, which is often available but rarely reliable. This extension works regardless of whether one, all or none of the persons in the household carry this sensor, and does not cause the tracking to fail should it be forgotten or otherwise not carried.

5 Implementation

Main goal of this work is to separate data of multiple persons moving in a space monitored by binary sensors. To accomplish this, we present

- a multi-target tracking algorithm optimized to separate of multiple targets in low-resolution data of ambient sensors
- an unsupervised learning approach to identification of smart home residents based on ambient sensor data and
- an extension of the multi-target tracking algorithm by an activity recognition model based on data from optional body-worn sensors for improved tracking and identification.

This chapter describes the technical details of the implementation of the multi-target multi-hypothesis tracking algorithm, the activity recognition and identification algorithms as well as the technical infrastructure required.

5.1. Architecture

The architecture of the completed system can be separated into three parts:

- the infrastructure (hardware and software) required to record, store and process events from ambient and body-worn sensors,
- the multi-target, multi-hypothesis tracking system and
- the activity recognition system.

5.1.1. Infrastructure

Many components and features of the system are predetermined by what hard- and software and other requirements smart-home and care monitoring systems bring. For example, sensors are often retrofitted, such that battery-powered and radio-transmitting sensors are often used instead of wired solutions. Figure 5.1 shows the main hardware and software components of the study conducted in the IDEAAL living lab, in which both ambient and body-worn sensors were used. It shows how multiple sensor technologies and smart home platforms can be

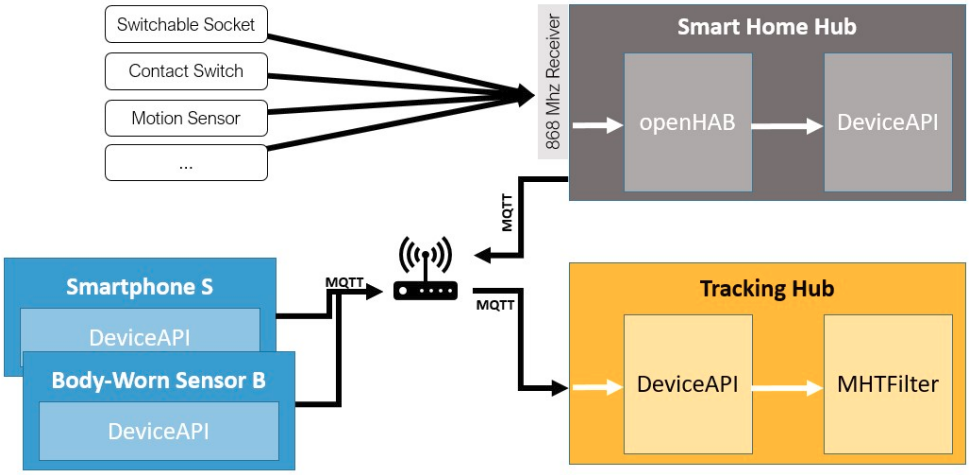


Figure 5.1.: Architecture diagram for the study conducted in the IDEAAL living lab

used for the tracking system through their integration with *openHAB*[1]. The sensor events are forwarded fed into the tracking algorithm via Message Queuing Telemetry Transport (MQTT). Thanks to MQTT, openHAB, the body-worn sensor data collection and the tracking algorithm can run on separate devices. The data arrives timestamped and is synchronized before updating the tracker.

The remainder of this section describes the technological and practical requirements that drove the design of the system.

5.1.1.1. Design

Client-Server Architecture Most smart-home platforms follow the client-server model structure, in which a central computer is installed which acts as the receiver for all sensor data. Some products, such as *Plugwise*[2] allow for individual devices to be connected to each other, forming a *mesh network*. Eventually, however, all sensor data is forwarded to a central "hub" or "gateway" to store or process.

Given that a multi-target tracking algorithm requires significant processing power, and as such energy, we assume a central computer that receives all sensor data and is able to process them in a timely manner.

Hardware Requirements The tracking algorithm is built with the intention to make a minimal amount of assumptions on the nature of the data it is receiving. Specifically, it is processing sensor data in which each datum represents evidence of human activity. The most representative example of such data is that of a passive-infrared motion sensor: it registers a change of infrared light across an array of sensors, then triggers a binary signal indicating motion.

Data from sensors with higher resolutions (laser scanners, cameras, radar) can still be used:

[1] https://www.openhab.org lists more than 360 add-ons at the time of writing.
[2] https://www.plugwise.com

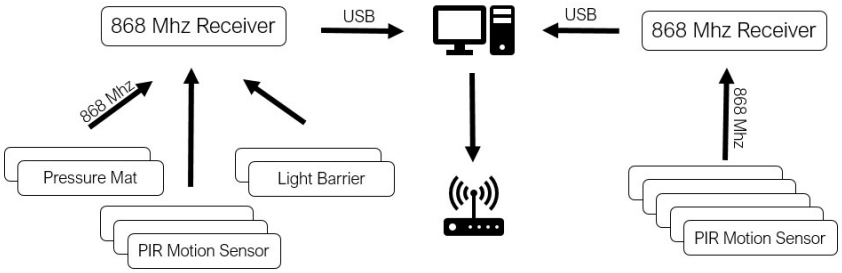


Figure 5.2.: Example client-server architecture diagram, as implemented in project `Cicely` [111]

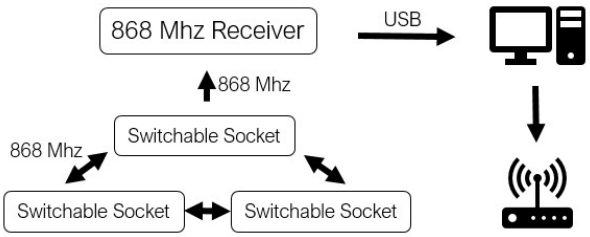


Figure 5.3.: Example mesh network architecture diagram, as used in project GAL-NATARS [59]

the data can be discretized and needs to be binarized in order to be usable in the tracking system.

The identification task could be achieved with specialized hardware. Ready-made solutions such as RFID-tags and -readers, facial recognition cameras and other products offer near-perfect identification, but require specialized and often expensive hardware. It is the explicit purpose of this work, to achieve identification without the help of identifying sensors.

5.2. Target Tracking

The tracking algorithm is based on the description of *hypothesis-oriented* multi-hypothesis tracking of Samuel S. Blackman [10]. The implementation is further based on the assumption that the sensor data is binary, meaning a sensor signal signifies either observed activity or absence of activity. Examples of such sensors are contact switches, passive-infrared motion sensors and light barriers. Main reason for this assumption is that many widespread smart home sensors provide binary data, so previously installed sensors do not need to be replaced.

5.2.1. Tracking on a graph

In order to create a space in which multiple targets can be tracked, we represent the sensors' sensing areas and their neighborhood relationships as a graph. In this graph, the sensor areas are the nodes. Two nodes are connected by an edge if it is possible to trigger the sensors subsequently (e.g. walk from the sensing area of one motion sensor to another) without triggering a third sensor in between. [111]

Definition 1. *A **sensor graph** of sensors $s_1, \ldots, s_N$ is graph $G = (V, E)$, where $V = 1, \ldots, N$ and $(u, v) \in E$ if and only if there is a path from u to v that does not cross another sensing area.*

This way, a single person can only trigger a sensor that is a neighboring sensor of the previously triggered sensor; in other words, a node that is connected to the previous node by an edge. Oh and Sastry [66] call two neighboring sensors "passage connected".

5.2.1.1. Generating a Sensor Graph from Data

As we saw, a great number of works cover the derivation of sensor network topologies from binary and higher resolution sensors [44]. These are also applied to or derived from medical or care-related applications. Most solutions, however, require a controlled initialization in form of supervised model training [96, 78] or detailed data about the monitored space, such as spatial distances between sensor areas or associations of sensors to rooms [23, 81].

Requirements such as having a blue print of the home or weeks of labeled data for supervised learning of models represent a significant obstacle for a larger adoption of technical care support systems. They burden the user with significant expenditure of time and money or expectations of technical know-how. Systems built on data provided at installation time also require maintenance for the inevitable cases where a sensor fails or is moved. In the worst case, the previously recorded data is then useless and setup must start anew. To avoid burdening the user, care support systems should be simple to install, require little to no time to set up, are ideally self-maintaining and rarely require human intervention.

Using previously recorded data, we can generate a graph by observing pairs of sensors that were triggered consecutively. Both the frequency of and the time difference between sensor events then gives us an understanding of the spatial arrangement of the sensors or their sensing area. [108]

In the theoretical case of a single-person household, in which there are no overlapping sensing areas, no visitors and no sensor failures, a list of chronologically sorted sensor events could simply be converted into a graph by adding an edge for any two sensors that trigger consecutively. In practice, however, there are many reasons why this approach would create erroneous edges, such as when there are multiple people present that simultaneously trigger sensors that have no neighborhood relationship or when a sensor event transmission is delayed due to network issues [57]. [108]

Depending on the application, we can generate a directed and weighted graph. In most cases of tracking in a smart home, an undirected graph is sufficient because there are few instances of routes that can only be covered in one direction. It may be useful, however, to weigh the edges based on some metric derived from the recorded data. For example, we might incorporate the information that a person is more likely to enter the living room than open the front door after entering the hallway. Such behavior would be represented by a larger amount of events by the motion sensor in the living room than the contact switch on the front door after an event by the motion sensor in the hallway. By counting the number of subsequent events

per neighboring sensor for each sensor, we can weight the outgoing edges for this sensor by their relative frequency:

Definition 2. *Let* $NB_i = j \in V : (i, j) \in E$ *be the set of all neighboring nodes of* i*. Then* $\sum_{j \in NB_i} p_{ij} = 1$ *is the sum of all transition probabilities for node* i*.*

Figure 5.4b shows the sensor graph for parts of the living lab at the Center for Advanced Studies in Adaptive Systems (CASAS) at the Washington State University depicted in Figure 5.4a. The graph is based on two months of activity data of two residents. The more frequent two sensors are triggered consecutively, the higher the weight of the edge between them and the thicker the line in the figure. As we can see, the actual neighborhood relationships all have large weights, but many edges exist that do not constitute actual neighborhood relationships.

Whenever more than one person is present, two sensors can be triggered consecutively that are not neighboring sensors. The suggested method of generating a sensor graph would then result in incorrect edges in the graph. However, in one-person scenarios the generated graph could provide a *more* precise graph than a manually constructed one: if two non-neighboring sensors are often triggered by one person, there might be an underlying explanation such as another sensor being unreliable. In such a case, this *virtual* neighborhood relationship will be picked up by the data and an edge between the otherwise not neighboring sensors is added. This way, the generation of a sensor graph from data may be more precise than its manual construction.

The `GraphUtil` class implements several functions that help convert sensor logs into graphs (`generateRelativeOccurrenceGraph`, calculating graph weights (`normalizeIncomingWeights`, `normalizeWeights`) and importing and exporting graphs (`saveToGraphML`, `saveToDOT`).

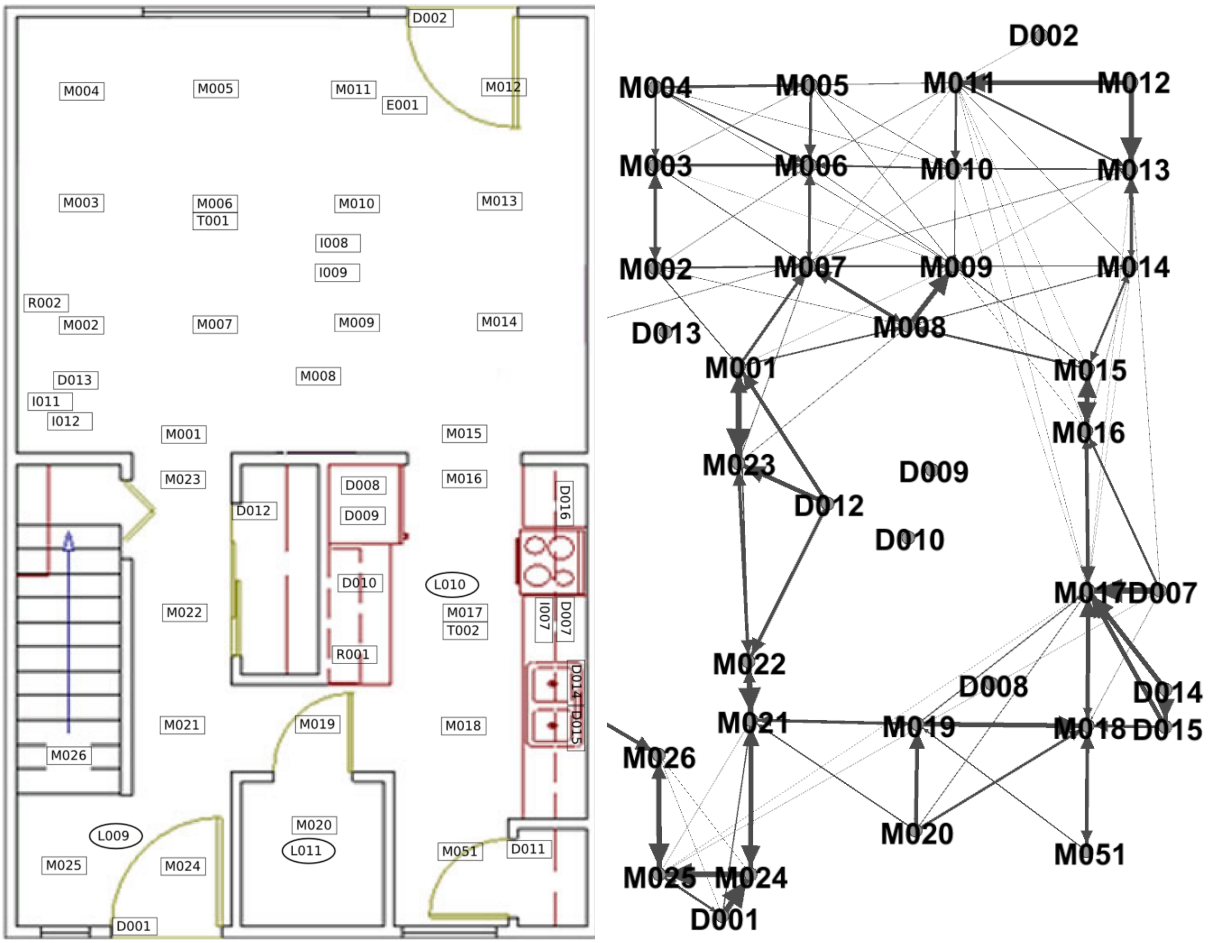
5.2.1.2. Constructing a sensor graph manually

A sensor graph can also be constructed manually. For this, it is necessary to have at least rough knowledge of the sensors' sensing area. Once these are known, it is easy to derive which sensors can be reached from each other. Using `JGraphT`'s `DirectedWeightedPseudograph`, we simply need to instantiate the graph, then add nodes and edges. Listing 5.1 shows how a graph of five sensors is constructed.

If it is known that a sensor is unreliable, it makes sense to also add edges between its neighbors. If it is clear that this sensor will often not be triggered, adding an edge between each of its neighbors is a way of adding this information to the tracker.

5.2.1.3. Non-binary sensor data

Main reason for the assumption that the sensor data is binary is that many widespread smart home sensors provide binary data. Additionally, sensors providing binary data often have little power consumption and therefore require little maintenance and can be used wirelessly. However, it can also be argued that any sensor whose data can be broken down into a binary



(a) Blueprint of a living lab with designated sensor installation locations for a study with two residents. Source: Cook et al. [15]. Printed with permission

(b) Directed, weighted graph based on data recorded in the same location. Line width represents average transition times

Figure 5.4.: Blueprint of a living lab with sensor locations and corresponding sensor graph

```java
DirectedWeightedPseudograph<String, WeightedEdge> graph =
        new DirectedWeightedPseudograph<String, WeightedEdge>(WeightedEdge.class);

// Nodes
graph.addVertex("knxMotionBathroom");
graph.addVertex("knxMotionCorridor");
graph.addVertex("knxMotionBedroom");
graph.addVertex("knxMotionLiving");
graph.addVertex("knxMotionKitchen");

// Loops
for (String node : graph.vertexSet()) {
    graph.addEdge(node, node);
}

// All other edges
graph.addEdge("knxMotionBathroom", "knxMotionCorridor");
graph.addEdge("knxMotionCorridor", "knxMotionBedroom");
graph.addEdge("knxMotionCorridor", "knxMotionLiving");
graph.addEdge("knxMotionLiving", "knxMotionKitchen");

return graph;
```

Listing 5.1: Example Java code for constructing a simple sensor graph.

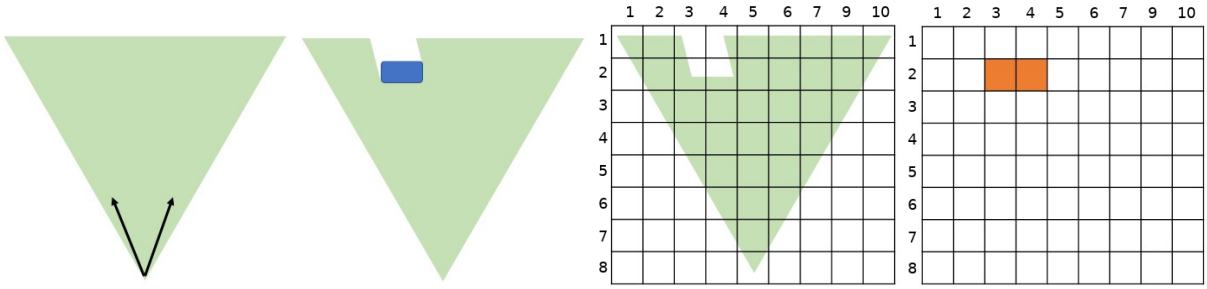
signal can be used. For example, a laser scanner provides two-dimensional data in the form of multiple distance measures across one axis. By discretizing the data, we can convert the data into a two-dimensional binary grid in which changes in the distance measure indicate activity in that specific cell of the grid. Figure 5.5 shows the scanning range of a laser scanner from above and how it can be discretized. When a person enters the scanning range, the cells that represent the x- and y-coordinates of the affected range report a binary activity signal.

5.2.2. Multi-Hypothesis Tracking

Each person in the target space is represented by a discrete Bayesian filter over the nodes of an unweighted, undirected graph (cf. Section 5.2.1), where the sensor areas constitute the nodes and the edges their spatial adjacency. Such a graph can be approximated by a path planning algorithm and a floor plan, if available [34], generated from prerecorded data [108] or constructed by hand.

5.2.2.1. Tracking of individuals

Although it might sound straightforward to track an individual through their home based on sensor data, there are various factors that can interfere with a flawless tracking: sensors are subject to noise, may fail to detect activity or detect irrelevant activity, such as from pets or through the window. Many motion and home automation sensors exhibit another source of



(a) No target (b) Target (c) Data discretized (d) Target location discretized

Figure 5.5.: Illustration of two-dimensional laser scanner data being discretized. Figure 5.5a shows the device at the bottom, scanning an even wall "in front" at a 60 degree angle. Figure 5.5b shows the scan being perturbed by an obstacle. Figure 5.5c shows where, on an 8x10 grid, this data falls. As we can see in Figure 5.5d, one or several targets at locations $(2, 3)$ and $(2, 4)$ can be derived from this discretization.

noise due to duty cycle regulations: the sensors do not measure or report measurements for a specified amount of time after triggering because their communication channel is only allowed to be used 1% of the time per hour. This "recovery" period can last from a few seconds to several minutes, resulting in missed movements and breaking the continuity of measurements of a motion track.

To estimate the location of a person, we model it as a probability distribution over all nodes in the graph based on a Bayesian filter. This may help to calculate a more accurate location estimation, especially when dealing with overlapping sensor regions, which is common with passive-infrared motion sensors mounted on walls or ceilings or any other sensor with a large sensing area. More importantly, however, it helps the tracker recover more quickly when a noisy measurement is assigned to individuals' tracks. Lastly, it is possible to use a *weighted* graph generated from previously recorded data instead of the manually constructed, unweighted graph to better reflect individual behavior by using transition probabilities between sensors as edge weights (cf. Marikanis et al. [58]). A Bayesian filter then updates based on these personalized edge weights; no further rework is necessary. [110]

The state of a target is derived from its associated sensor events only[3]. The likelihood calculation is analogous to Formula 3.1, whereby

- the sensor model $p(z_t|X_t = x_t)$ is defined by two constants, `hitProbability` and `missProbability`, that describe the probability of a sensor firing and not firing during activity, and
- the motion model $p(X_t = x_t|X_{t-1} = x\prime)$ is based on the distance on the graph between the sensors triggered during the current update and the last event associated with this target.

[3]cf. Wilson's work described in Section 3.1, in which the state is a combination of the target's location *and* activity status.

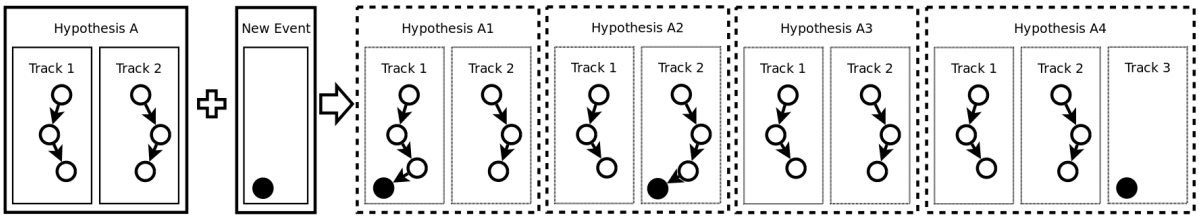


Figure 5.6.: Hypothesis formation overview

Whether an event is assigned to an existing track or spawns a new one – that is to say, how many individuals are assumed to be present – depends on the *belief* of the assignment of the event to *any* existing track: if a new measurement causes the current data to be more or less likely when assigned to more or fewer tracks than up to this point, a filter is created or discarded. As stated by Blackman [10], updates can be bundled into larger updates within reasonable time frames. In our case, an update occurs for each new sensor event.

5.2.2.2. Multi-Target Tracking

To distinguish activity of two or more persons in an area monitored by binary sensors, the activity must be spatially separable on the graph. Ideally, there is an inactive sensor between two persons. This concept has been thoroughly described by Oh and Sastry [66]. Since overlapping motion paths and activities are common in a household with two or more residents, we implemented a variant of the multi-hypothesis tracking algorithm as described by Blackman [10].

Similar to the common track pruning approach described in Section 3.1.2, we implement a method of trimming hypotheses as a whole. Unlike the track pruning, we prune the hypotheses and discard sensor data when *no association* can be made. The main motivation for this is that many applications do not require a target to be tracked continuously, but rather reliably across short periods. For example, in order to determine changes in mobility of a person by measuring the time it takes a person to walk between two sensors over several months, it is sufficient to track this person between two or a few sensors once a day. Also, if the data is used to support care-related or medical decision making, it makes sense to discard data that we cannot be certain of (data associations with low probabilities) rather than including them in the analysis.

Instead of pruning or clustering tracks to limit the number of hypotheses, this implementation uses a temporal *window* in which data associations need to be made. The window size is defined by the maximum number of updates (i.e. sensor events) that will be fit into a hypothesis. If the algorithm does not find a single, dominating hypothesis until the window size is reached, data from the beginning of the window is dropped as `not associated`. Unlike in the particle filter approach described in Section 3.1, the sensor event assignments do not need to be sampled because the state space is cut off in time by the window size.

Procedure Listing 5.2 shows the general sequence of steps that are involved in processing sensor events into hypotheses. After new hypotheses are generated (lines 6-9) for a sensor event, a `gating` function filters all event-to-track associations in which the event is more than the maximum distance away from the last event in the track (lines 12-13). The maximum distance can be chosen freely but is usually a function of the reliability of the sensors and the size of their sensing area. In the case of tracking on a graph, the gate size is a function of the distance of two nodes and the weights of the edges in between. In our evaluation, all edges of the graphs are weighted with 1 and the gate size is set to 2, meaning that if an event is further than 2 steps away from the previous event in the track, the association of the event to this track is not considered. Afterwards, hypotheses are filtered based on confidence, noise ratio and similarity (lines 15-16). The remaining hypotheses will be the basis of newly generated hypotheses when the next sensor event arrives. This procedure is performed until a single hypothesis remains (lines 18-20) or the window size is reached. In the former case, the hypothesis is accepted and stored, the underlying Bayesian filters updated, and the window size reset. In the latter case, all hypotheses are evaluated. If no single, dominating hypothesis can be found (lines 22-25), the first event in the window is discarded, the underlying filters reset and the remaining data in the window recalculated. [112]

The sensors are treated like the "type 2" sensors described by Reid [71], meaning we only consider "positive reports", i.e. events that signal human activity[4]. For each sensor event e_r of a sensor r, a new hypothesis based on all previously existing hypotheses is created (see Figure 5.6, in which e_r is

- considered noise and discarded,
- used to update one of the existing filters, or
- assigned to a new filter.

A hypothesis j at time t contains the targets' states x_t as defined by Bayesian filters and implemented in the `BayesianGraphFilter` and lists of associated sensor events $\theta_{1:t}^{(j)}$ for all targets as implemented in class `MovementPath`:

$$h_t^j = \{x_t^{(j)}, \theta_{1:t}^{(j)}\} \tag{5.1}$$

For the evaluation of a hypothesis, we take advantage of the fact that the number of targets in a domestic environment is mostly static. Hypotheses are evaluated according to Equation 5.2. This formula takes into account the probabilities from the Bayes filters for all targets in a hypothesis and weighs it against the number of *expected* targets.

$$eval(h_t) = \frac{\sum_{i=1}^{n} p(X_t^{(i)} = x_t | z_{1:t})}{\frac{n^2+m}{m+1}}, \tag{5.2}$$

[4]Many smart home sensors also send *OFF* events that inform of the end of a sensor's down time

where n is the number of paths (= targets) in h and m is the expected number of targets in the sensor space. Informally, the equation sums up the probability of all tracks in a hypothesis, then normalizes it by dividing by the number of tracks. The expected number of tracks is added to reward hypotheses with tracks close to the expected number of people present. [112]

Postponing the association decision until a predominant hypothesis exists is particularly useful in a low-resolution setting, where individuals may occlude each other in sensor readings for any period of time. Since the number of possible combinations increases quickly with the number of sensor events, this procedure is limited by the amount of hypotheses the computer can retain in memory. To accommodate this, we introduce a parameter referred to as `window size`: the maximum number of updates before (a) a hypothesis is accepted or (b) data is discarded because no decision could be made based on the available data. While a larger window will likely result in greater tracking precision, it might also cause

- more errors in individual hypotheses, because the false association of a single event usually carries subsequent association errors and
- fewer total associations, because data from the beginning of the update window is discarded if no predominant hypothesis is found. In a larger window, this might lead to more discarded data.

The first evaluation in this work will test both of these claims.

Hypothesis-oriented MHT In 1979, Donald B. Reid published the original work on multi-hypothesis tracking [71]. It was designed to work on radar data, a two-dimensional continuous space. In the work presented here, the target space is discrete (nodes on a graph), and targets and their locations are stored as a probability distribution over the space using Bayesian filters.

Three differences between Reid's original work and our approach should be mentioned here. First, Reid defines a *type 1* and *type 2* sensor. We only deal with *type 2* sensors, meaning the sensor model used in this work expects *positive reports* only, which in turn means that only sensor data reporting activity is considered. Furthermore, tracks are updated *per hypothesis*, as opposed to the *hypothesis-oriented MHT*, where each track is generated and filtered individually. Thus, hypotheses are not constructed from compatible tracks, but the list of all possible hypothesis and update combinations. For a detailed discussion of track- and hypothesis-oriented multi-hypothesis tracking, see Blackman [10]. Lastly, the tracker is updated every time a sensor reports activity. Because of this, and the fact that the given state space is discrete, computational complexity is reduced.

Filtering For a specific number of possible targets, we can calculate the number of possible hypotheses – i.e. the number of possible combinations of events to a fixed number of tracks – using the *Stirling number of the second kind* [45]. It describes the number of ways to partition a set of n objects into k non-empty subsets and is defined by

```
function update(List<Hypothesis> hypos, String sensorEvent) {
    if (hypos.length == 0)
        hypos.add(new Hypothesis(""));
    List<Hypothesis> h0;
    // Step 1: Generate hypotheses
    for each (Hypothesis hCurr in hypos) {
        List<Hypothesis> children = generateChildrenHypotheses(hypos, sensorEvent);
        h0.addAll(children);
    }
    for each (Hypothesis h in h0) {
        // Step 2: Gating
        if(distance(getLastEvent(h), sensorEvent) > gate_size)
                h0.remove(h);
        // Step 3: Filtering
        else if(anyFilterApplies(h))
                h0.remove(h);
    }
    // Step 4: Evaluation
    if(h0.length == 1) {
        // Accept if single hypothesis remains
        acceptHypothesis(h0.get(0));
    } else if(h0.length > 1) {
        // Accept dominating hypothesis
        h0 = h0.sort(confidence, "descending");
        if(h0.get(0).confidence > h0.get(1).confidence + DOMINATE_THRESHOLD)
                acceptHypothesis(h0.get(0));
    }
}
```

Listing 5.2: Pseudocode for processing of a single sensor event update.

$$\binom{n}{k} = \frac{1}{k!}\sum_{i=0}^{k}(-1)^i\binom{k}{i}(k-i)^n \tag{5.3}$$

For an unspecified number of targets, the number of possible hypotheses follows *Bell's Number* [9]:

$$B_n = \sum_{k=0}^{n}\left\{ {n \atop k} \right\} \tag{5.4}$$

For the evaluation, we tested window sizes between 10 and 20; higher values proved too time consuming to test out. For a window size of 20, the number of possible hypotheses exceeds 474 trillion $> 4.74 \times 10^{13}$. Since it is not possible to retain all hypotheses in memory, we apply a number of filters to remove the least likely ones and remove unnecessary ones (for example, hypotheses for which a more likely hypothesis with a higher signal-to-noise ratio exists) to reduce the overall number of hypotheses. There are various metrics by which we can eliminate a large number of hypotheses before or after they are evaluated. The filters implemented for our evaluation are described in Section 5.2.2.3. The thresholds were chosen based on personal observations, but some general suggestions can be made. For example, the less powerful the hardware on which the algorithm runs, the stricter should the filters be. If the hardware allows, the filters can be disabled altogether.

Figure 5.7 shows – on a logarithmic scale – an example for the number of hypotheses for eleven events and how different filters affect the number of hypotheses retained. As we can see, the gating does not have a significant effect, though its impact varies with the size of the gate. Confidence and similarity filters lower the number of hypotheses by several magnitudes. The filters are:

Gating Removes hypotheses where the the previous event is far away from the current event.

Confidence Removes hypotheses that are below a fixed or relative (to the most likely hypothesis) probability threshold.

Similarity A hypothesis is removed if a similar hypothesis with a higher probability exists. The similarity of two hypotheses is defined by the sum of the Levenshtein distance [49] of their tracks.

Combined A combination of all of the above.

5.2.2.3. Software Implementation

This section summarizes the interactions of the core components of the multi-hypothesis tracking implementation and describes each component in detail.

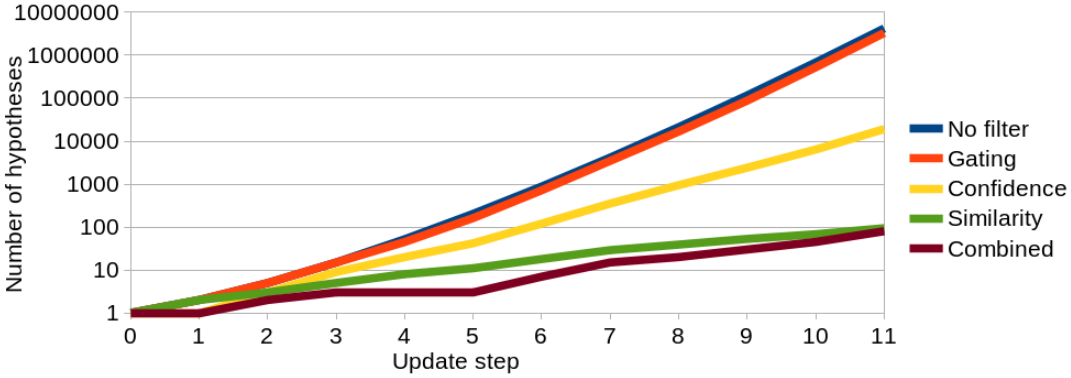


Figure 5.7.: Number of hypotheses across window size with various filters

Overview The core logic of the algorithm is implemented in the `MHTFilter` class. To set up the tracking, an `MHTFilter` must be instantiated. The constructor requires a `DirectedWeightedPseudograph`, an implementation of a graph that allows multiple weighted edges between nodes, provided by `JGraphT`.

New sensor events are added to the tracking by calling the MHTFilter's `update` method. At each update, multiple new `PathHypotheses` are created based on the pool of currently maintained hypotheses.

For each path, a `PathHypothesis` maintains a `Filter` and a `MovementPath`. The former stores all data necessary for calculating the location probability of a target, such as the prior and posterior probability for recursive Bayesian filtering and a sensor model (*hit* and *miss* probabilities for correct and incorrect sensor activations). The latter serves mainly as a data store for the raw sensor events and provides metrics such as a similarity measure and a count of the number of sensor events classified as noise.

The `BayesianGraphFilter` is the primary Bayes' filter implementation used for the first evaluations. We will later extend the filter concept by overriding the activity model, which will be used later to evaluate whether body-worn sensors help improving the tracking precision.

All classes are implemented as a Maven[5] package in Java using OpenJDK 11[6].

MHTFilter The MHTFilter is the core of the tracking system: it is the place for data preprocessing, such as filtering sensor events or manipulating them for faster processing; it provides an `update` method which generates new hypotheses out of a pool of previous ones; it implements the evaluation function for hypotheses, it maintains the pool of hypotheses, and therefore all targets and paths.

The creation of new hypotheses is parallelized in a private class called `HypoRunner`. For each hypothesis from the hypothesis pool, a new `HypoRunner` is instantiated that will create at least three copies of this hypothesis for each of the events in the update:

1. a copy for each of the tracks being associated with the new event,

[5]https://maven.apache.org/
[6]https://openjdk.java.net/

2. a copy where the new event constitutes the beginning of a new track and
3. a copy where the new event is discarded as noise.

After collecting all children hypotheses from the `HypoRunner` instances, they are subjected to a number of filters designed to minimize the size of the hypothesis pool. The following list shows the filters implemented as well as the values set for our evaluation (probability and similarity values are given as percentage values). It should be noted, however, that neither the type of filter nor its parameters are in any way fixed other than by the specifications of the hardware the algorithm runs on.

- Discard all hypotheses that do not reach a global minimum probability threshold (10),
- Discard all hypotheses that have a probability smaller than the probability of the most likely hypothesis divided by a factor (3),
- Discard all hypotheses where the new event is outside of the "gate",
- Discard all hypotheses that have a probability lower than the probability of the most likely hypothesis minus a subtrahend (70),
- Discard all hypotheses that are *similar* to a more likely hypothesis and contain more noise (`similarityScore` > 95).

If exactly one hypothesis remains, it is accepted. That means, the track data in it is stored and the *window size* reset. If more than one hypothesis remains, the `checkForConsensus` method will compare the track data in all remaining hypotheses chronologically and mark all events as accepted that are associated to the same path in all hypotheses.

There are various parameters through which an MHTFilter can be customized that affect the tracking behavior:

gating_max_jumps The size of the gating window, given by the maximally allowed distance on the sensor graph between the current sensor event and the last.

max_hypotheses The maximum number of hypotheses to maintain in order to avoid errors due to lack of memory.

min_diff_to_accept If a hypothesis "dominates", i.e. it is significantly more likely than any other hypothesis, it can be accepted if its probability is this much larger than the next likely hypothesis'.

min_global_confidence Discard all hypotheses that have a confidence below this value.

similarity_threshold If two hypotheses score this much or higher on a similarity score, discard one if it is less likely and contains more noise.

PathHypothesis A `PathHypothesis` maintains all data necessary to maintain, evaluate and update a hypothesis. This mainly refers to `ArrayList`s of `Filter`s and `MovementPath`s, but also a reference to the underlying sensor graph to calculate distances. Note that the use of `ArrayList` allows for quickly adding and removing of targets. Each PathHypothesis contains a reference to the hypothesis it was derived from, and its ID will be generated from its "parent hypothesis", so that it is easy to retrace how the hypothesis was formed.

MovementPath The `MovementPath` class is a data store for sensor events that are combined into a "track". It also provides utility functions and metrics, such as the amount of noise in the track or calculating the *Levenshtein distance*[7] between two tracks.

BayesianGraphFilter The core functionality of a `Filter` is to calculate and store the location probabilities for a target. The target states are derived from the sensor graph introduced via the `MHTFilter`. The `BayesianGraphFilter` calculates location probabilities using Bayes' theorem,

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)} \tag{5.5}$$

where $P(x|y)$ is the the probability of the target being at location x given an event at sensor y (referred to as the *posterior probability*), $P(y|x)$ is the probability of a sensor y being triggered given the target location x (referred to as the *sensor model*), $P(x)$ the *prior location probability* and $P(y)$ the probability of sensor y triggering regardless of target location (referred to as *evidence*). Commonly, the priors are uniformly distributed at initialization.

In the `BayesianGraphFilter`, the sensor model is defined by a `hit` and `miss` probability, which define the probability of the sensor reporting an event given a target has entered its sensing area and the probability of the sensor *not* reporting an event given a target has entered its sensing area. For the evaluation, these values are set to static values of 0.96 and 0.02 based on experience with the hardware used.

An overview of the main components and their interactions is depicted in Figure 5.8.

5.2.3. Software Dependencies

The Maven package, which contains the tracking algorithm, depends on the following external packages:

Log4J 2.7 Log4J is a logging framework that helps logging program output. It is easily configurable using configuration files and supports parallel output of differing levels to different locations, including console, local or remote files.

[7] The *Levenshtein distance* (after V. Levenshtein [49]) is defined as "the minimal number of insertions, deletions and replacements to make two strings equal". [63]

Figure 5.8.: Core classes of the ambient-only MHT implementation. C denotes constructors, S static members and F final members

Guava 21.0 Guava is a multi-purpose Java library containing numerous collection types and graph functionality. Guava is used for its `TreeMultimap`, a mapping collection, to sort hypotheses by their confidence between tracking updates.

JGraphT 0.9.2 JGraphT is "a Java library of graph theory data structures and algorithms"[60] and provides the core data structures and functions for the graphs used in the multi-target tracking library. Most notably, it provides the `DirectedWeightedPseudograph` class, which implements a directed graph in which graph loops as well as multiple parallel edges are permitted and edges have weights. It is used to represent the sensor graph. It also provides algorithms to find the shortest path between graph nodes (`BellmanFordShortestPath`) and functions to import and export graph data (`DOTExporter`, `GraphMLExporter`).

HPPC 0.7.2 HPPC stands for *High Performance Primitive Collections* and implements high performance collections for primitive Java types (`LongArrayList`, `IntObjectMap`, etc.). It is used most notably for the *timed* multi-target tracking where updates are provided as arrays of sensor IDs (`integer`) and timestamps (`long`).

OpenCSV 2.3 The OpenCSV library is used to import and export sensor data from comma-separated value (CSV) files.

Apache Commons DBPC 1.4 DBPC stands for *Database Connection Pool* and implements fast and efficient establishment and termination of database connections. The multi-target tracking library uses it to store track and sensor event data as well as log messages.

MySQL Connector/J 8.0.16 The MySQL Connector is required to set up a connection to a MySQL database, which is used to store track data.

5.2.4. Data Flow

Commonly, a program utilizing the the tracking algorithm will go through the following steps before instantiating the `MHTFilter`.

Setup

1. Set up a connection to a sensor data source, such as a network connection to a smart home hub or a file,
2. set up filters for the sensor data, such that only sensor events relevant to trackable activities are forwarded to the tracker,
3. adapt the parameters, especially the filter thresholds, depending on the number, resolution and density of sensors, and

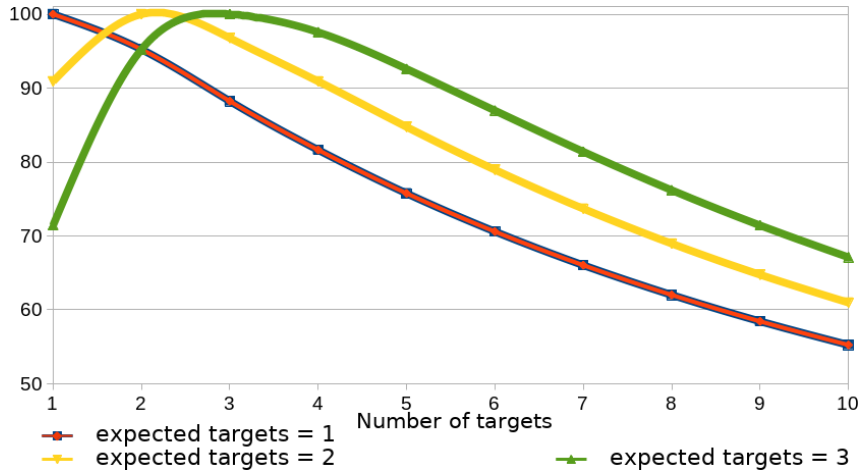


Figure 5.9.: Influence of number of expected paths on evaluation function

4. create an instance of a `DirectedWeightedPseudograph<Integer, WeightedEdge>`

Once the `MHTFilter` is instantiated, it can be fed with sensor data using the `update(long[] timestamps, int[] sensorIDs)` function.

Update The `update` method of the `MHTFilter` takes a list of timestamps and sensor IDs as input. In the case of an event-based tracker, calls to this function usually contain only one event. It is possible, however, to collect events over a period of time (often 10 seconds or more) and then batch-update the tracker.

Evaluation of hypotheses Equation 5.2 describes the hypothesis evaluation function. As a parameter, it takes the expected number of targets. In the case of domestic monitoring, this is the number of residents. In other words, it is the number of targets that is to be expected the majority of the time. As can be seen in Figure 5.9, the evaluation function has a maximum value of 100 for the number of expected targets. In our scenario of two expected targets, the highest possible probabilities are for hypotheses with one to four paths. At five or more paths, the probability slowly trails off towards zero.

Filtering hypotheses After evaluation, all hypotheses pass through an array of filters that help keeping the computation effort manageable. The filters implemented in this work are described in Section 5.2.2.2.

Preparation of next update Based on the number of remaining hypotheses, several steps may occur after filtering. If only one hypothesis remains, the track data is stored and the window size is reset. The Bayesian filters maintain the location information for all targets. If more than one hypothesis remains and the `window size` is reached, the Bayesian filters are reset, the first event in the window is removed and the tracks are recalculated based on the rest of the sensor events in the window.

5.2.5. A Note on the Cessation of Tracks

Since it is impossible to differentiate between absence, inactivity and presence in space not covered by sensors from sensor data alone, we must employ additional sensors or techniques to distinguish between these situations. For this, we have a look at three common applications of tracking in home scenarios and discuss possible solutions.

Application 1: Lighting Automation In order to predict which way a person moves and what light they might switch, it is sufficient to track them over short periods. Once a person becomes stationary, the lighting is not expected to change. Furthermore, it is mostly irrelevant if a person is opening the front door to leave the house or the receive a package: actions to be taken are based on activity data of only a few seconds prior. Tracks may quickly fade (decrease of probability over time until they drop out) and no further sensors are necessary.

Application 2: Presence Count It is straightforward to derive the number of people present from a multi-target tracking hypothesis by counting the number of targets in it. However, most of the time the algorithm does not contain a single hypothesis, but several. One way of estimating the presence count is taking a weighted average of the number of targets in the hypotheses based on the hypotheses' evaluation. The fading of tracks should be disabled or slowed, so that motionless persons' tracks do not disappear[8]. It must be noted, however, that the tracking algorithm is a lot less likely to recover from tracking errors if fading is disabled, because "abandoned" tracks due to tracking errors can remain in the system indefinitely. In order to be able to *end* tracks when a person leaves the house, we must install a sensor on all exits, e.g. a contact switch on the door.

Application 3: Inactivity Alarm If we were to implement a system to raise an alarm if a person falls and is unable to stand up, no alarm should be raised if a person's track ends or pauses on their bed. Conversely, if that track ends or pauses anywhere other than a place of rest, an alarm should be raised. For this, the tracks may end on their bed or sofa, but should not end anywhere else. This requires either specialized hardware, such as pressure sensors in the bed, or an annotation of the sensors where a track may fade.

5.2.6. Examples

The following examples show the functioning of the algorithm based on scripted scenarios recorded in the living lab. Example 1 shows how the most likely hypothesis changes through a two-person scenario. Example 2 shows how the tracker recovers from noisy sensor data. Example 3 shows a similar example, but where the tracker fails to recover from noisy sensor data.

[8] depending on whether a sleeping person counts as "present".

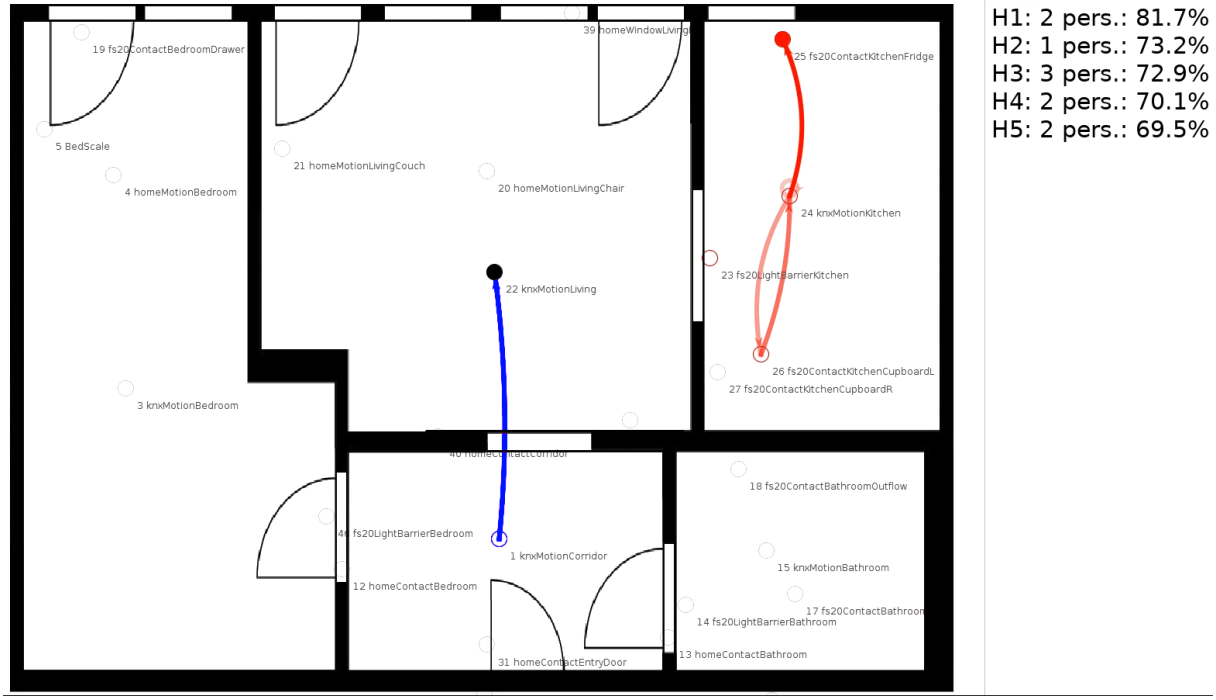


Figure 5.10.: Example tracking scenario visualized (current hypotheses right), step 1

5.2.6.1. Example 1

Figures 5.10, 5.11 and 5.12 shows the visualization of the tracking results while two people walk through the IDEAAL living lab [47], visualized as a blueprint of the lab with the included sensors (PIR motion sensors, light barriers and contact switches). On the left, there is the list of currently maintained hypotheses, with the last four transitions of each track of the most likely hypothesis depicted in the blueprint.

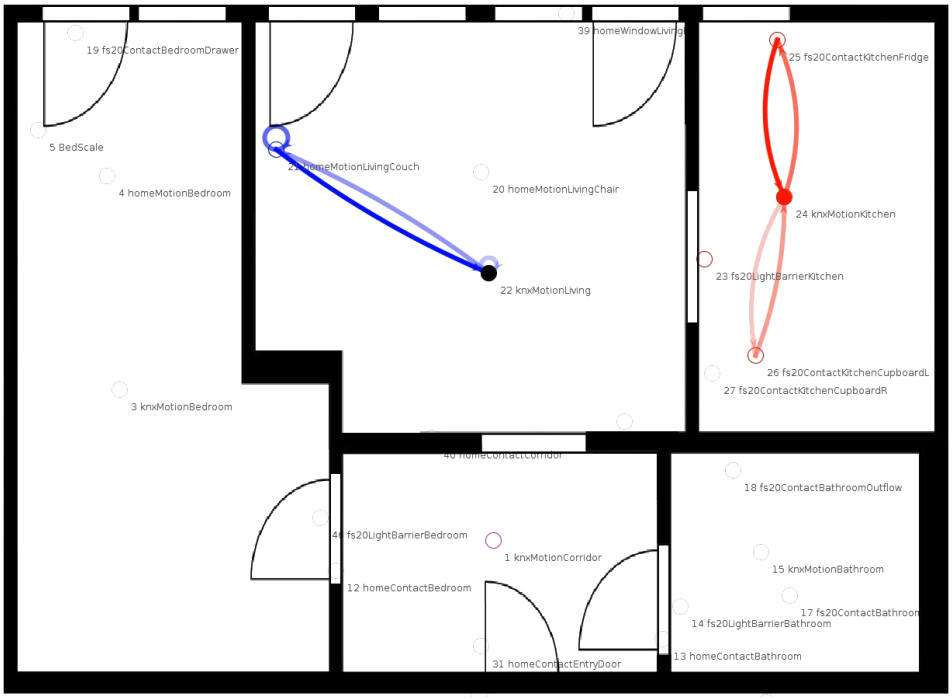
The two persons entered the lab with a slight temporal offset. The first person walked from the hallway through the living room into the kitchen. As the second person enters the living room from the hallway (Figure 5.10), five hypotheses are maintained, of which three include two tracks, one includes one track and one includes three. In the blueprint, the most likely hypothesis is visualized, which, initially, is the correct one. One event later, depicted in Figure 5.11, the most likely hypothesis contains only one track. This can be explained by the second person moving closer to the first one and the first person not moving at all during this time. The transition of the single target from the kitchen to the living room is not visualized, because the graph contains no edge between the last two sensors that fired. Three events later (one in the kitchen, two in the living room), the tracker has settled on the correct hypothesis, although the two participants are only separated by one sensor (`fs20LightBarrierKitchen`).

5.2.6.2. Example 2

In this example, a single person moves through the lab. Figure 5.13a shows that the most likely hypothesis is based on two targets, because, after walking from the kitchen (right) to the

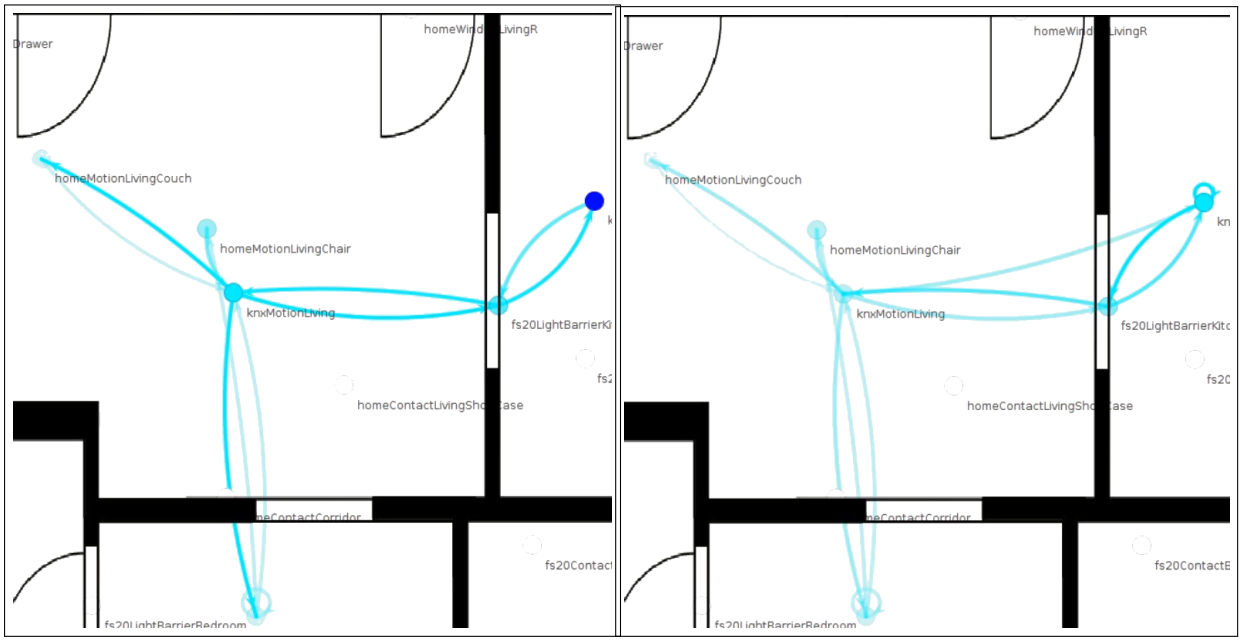
H1: 1 pers.: 88.4%
H2: 2 pers.: 85.3%
H3: 2 pers.: 80.0%
H4: 3 pers.: 76.2%
H5: 2 pers.: 74.8%
H6: 3 pers.: 73.3%
H7: 3 pers.: 73.3%
H8: 2 pers.: 71.8%
H9: 2 pers.: 71.8%
HA: 2 pers.: 70.8%
HB: 2 pers.: 70.8%

Figure 5.11.: Example tracking scenario visualized (current hypotheses right), step 2



H1: 2 pers.: 85.9%

Figure 5.12.: Example tracking scenario visualized (current hypotheses right), step 3



(a) Second example tracking scenario, step 1 (b) Second example tracking scenario, step 2

Figure 5.13.: Visualization of example tracking scenario

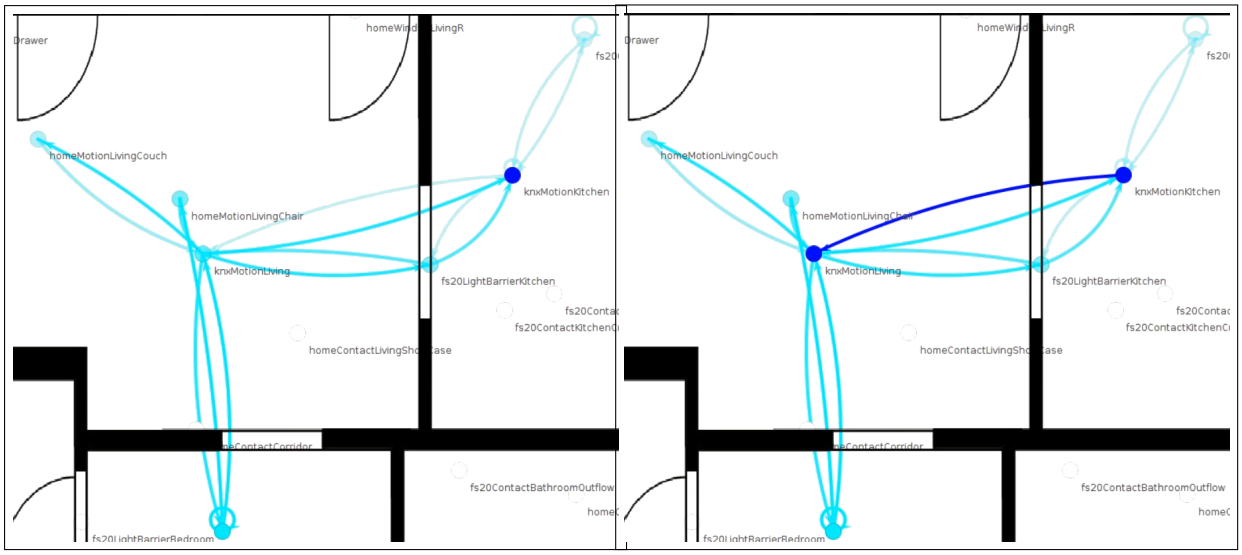
living room (left), the light barrier in the door does not trigger as the person walks back into the kitchen. Therefore, the tracker assumes a person at the `knxMotionLiving` and `knxMotionKitchen` sensors. Figure 5.13b shows that the tracker returns to the correct hypothesis after the next event at sensor `knxMotionKitchen`.

This example highlights another important aspect of tracking with binary sensors. The KNX motion sensors in this example have a wide range and are installed on the ceiling. This is common in smart home installations, which often rely on covering the largest possible area. As a result, it may be difficult to install sensors without sensor areas overlapping. Since the motion sensors have a large sensor area and the light barrier between living room and kitchen has a very narrow sensor area, it is possible that the person triggered `knxMotionLiving` without leaving the kitchen. In this scenario, die light barrier did not fail to trigger; instead, the underlying sensor graph is imprecise because it does not account for the possibility to trigger `knxMotionLiving` and `knxMotionKitchen` consecutively. Thanks to the multi-hypothesis tracking algorithm, however, the result is the same.

5.2.6.3. Example 3

For this example, the underlying sensor graph was adapted to include the possibility to move from `knxMotionLiving` and `knxMotionKitchen` without triggering `fs20LightBarrierKitchen`.

This example starts similar the the previous one: a person walks from the kitchen to the living room. Now a second person enters the lab through the front door (bottom center). Instead of spawning a new track at this position, the most likely hypothesis spawns a new path at the position of the first person and moves the first person to the front door. While the number of



(a) Two-person tracking error, step 1 (b) Two-person tracking error, step 2

Figure 5.14.: Two-person tracking error visualized

targets is correct, the previous data of the first person is now associated with the second person. If it weren't for the corrected graph, the next event would probably resolve the situation and correctly associate the two events in hallway and living room to the second person. However, due to the correction of the sensor graph, both persons now have the same distance to `knxMotionLiving`. The most likely hypothesis is the association of the subsequent event in the living room with the first person coming from the kitchen. From here, the tracker does not recover.

5.3. Identification

As described in Section 4.2, identification ideally happens without the use of complex or body-worn sensors that may interfere in the daily life of users. Based on the assumption that the data recorded with ambient sensors and processed into tracks by the multi-target tracking algorithm provide enough information to separate tracks into personal track clusters, or motion profiles, we label a track dataset and test three clustering algorithms to determine if and how a separation of residents can be achieved.

5.3.1. Data Preprocessing

Identification based on activity data assumes that people are distinguishable based on features and metadata of their activities. Observable differences in activity data could be based on: differences in daily routine, such as regular presence or absence on specific days or during specific times of the day; presence in specific areas of the house, such as individual bedrooms, an office or the kitchen; motion parameters such as gait speed; metadata such as temporal overlap of tracks.

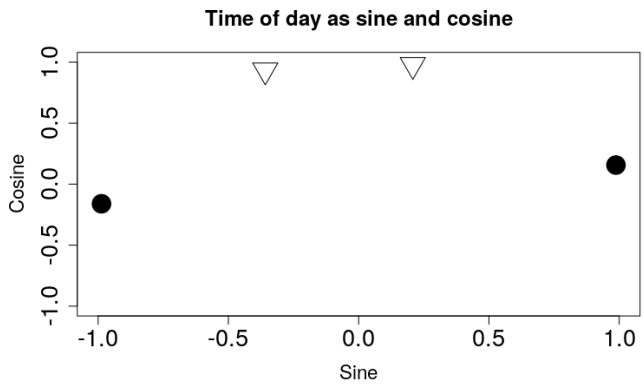


Figure 5.15.: Example for start and end times of tracks represented as sine and cosine. The tracks starting at 05:24 and 17:23 (represented as circles) are maximally apart while the ones starting at 22:36 and 00:48 (represented as triangles) are close.

The data stems from a tracking algorithm such as the one described in Section 5.2. The only requirement is that the activity data contains an ordered list of sensor identifiers and timestamps, representing motion. The output of the above-mentioned algorithm is stored in a database of tracks. These tracks vary in length, both temporally and in the number of sensor events, because tracks end after a person stops moving, or at least activating sensors. Furthermore, the length of tracks is dependent on the unambiguousness of the data, which itself depends on the number of people present, their spatial proximity, the range of each sensor and the number of sensors installed.

For the clustering and classification procedures, for each track we extract

- day of week,
- time of day,
- a binary vector describing which sensors were activated during the track and which were not,
- a binary vector describing which rooms contain sensors that were activated during the track and which were not,
- relative gait speed, calculated by the median transition time between all pairs of adjacent sensors.

So as to not lose the information about the cyclical nature of the time of day, we represent the time of day as sine and cosine. To do so, we calculate the minute of the day for the start (m_s) and end (m_e) of the track, then multiply each with $\pi/180$ to convert to radians. From this, we can easily calculate sine and cosine. Figure 5.15 shows an example of two tracks being maximally apart (12 hours) and two tracks being close across dates.

5.3.2. Supervised

First, we label the tracks produced by the multi-target tracking algorithm from the data recorded at the WSU CASAS living lab with IDs 1 and 2 for Resident A and B. Then, we use this labeled data to train a classification model: if a supervised learning algorithm is able to separate the residents with sufficient accuracy, we can also expect to be able to generate a model using the unlabeled and an unsupervised learning algorithm, in the hopes that the system can be installed and operated without human interference, long data recording, labeling and learning periods. If, on the other hand, the classifier performs poorly, it is unlikely that an unsupervised learning algorithm will be able to perform satisfactorily.

Track Classification For classification, we chose *WEKA*'s (Waikato Environment for Knowledge Acquisition, version 3.8.3 [39]) `J48`, a Java implementation of the C4.5 decision tree algorithm [70]. The tree is trained with ten-fold cross validation with default parameters (confidence threshold for pruning = 0.25, minimum number of instances per leaf = 2) on a track database comprising a total of approximately 11000 sensor events. Appendix A.1 shows the generated classification tree and related output of WEKA for the training and testing procedures.

The gait speed parameter introduces a significant number of additional features: one for each edge in the sensor graph, 95 in the WSU CASAS dataset. Visual inspection of the data shows that the set of gait speed values for most edges is highly scattered, thus not particularly useful or expressive. Therefore, we remove all gait speed features from this analysis. Based on a set of tracks without noise, 97.98% of tracks are correctly associated with another. Using the full dataset including noisy data caused by noisy sensor data as well as tracking inaccuracies, the classifier ist still able to correctly classify 73.2% of tracks. [113]

Classification Features To optimize the feature space for the unsupervised approach, we use WEKA's attribute evaluation feature to gain insight into the usefulness of the track features for their separation. `InfoGainAttributeEval` evaluates the usefulness of an attribute with regard to classification by measuring the information gain (IG) with respect to the class, which is then expressed as the *merit* of each feature. Merit is based on the order of selection of classification criteria of the algorithm. Since C4.5 uses IG as its feature selection measure, the merit is in turn based on the IG of said feature. Information Gain is defined as

$$IG(D, a) = E(D) - E(D|a) \tag{5.6}$$

where $E(D)$ is the entropy of dataset D defined as

$$E(D) = \sum_{i=1}^{c} -p_i log_2 p_i \tag{5.7}$$

where c is the number of classes in D and p_i is the fraction of instances of class i in D and

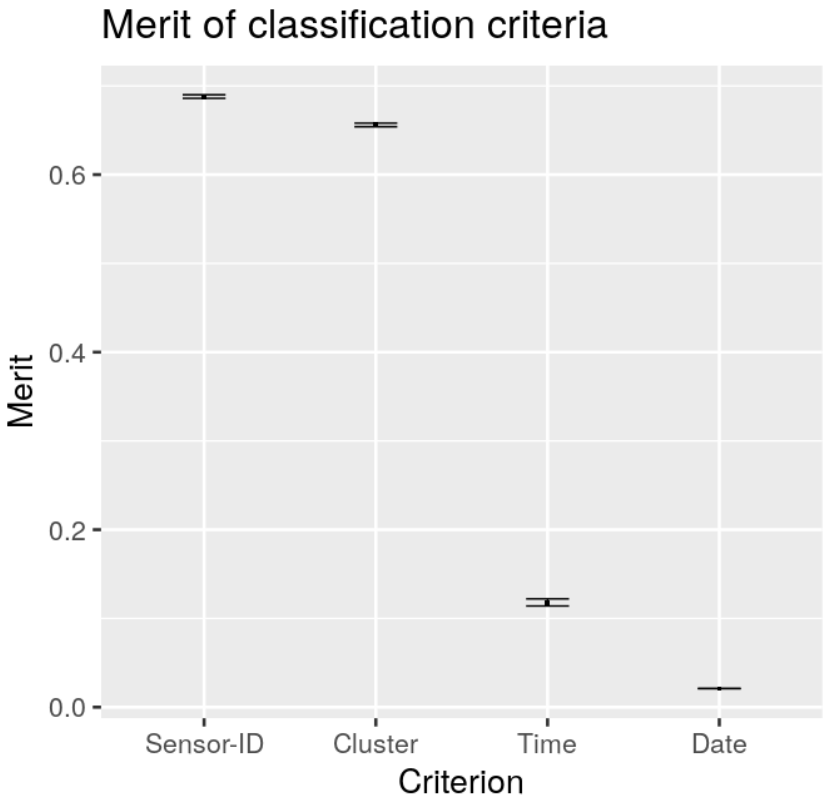


Figure 5.16.: Mean and standard deviation of merit of classification criteria during 10-fold cross validation

$$E(D|a) = \sum_{v \in vals(a)} p_a(v) E(S_a(v)) \tag{5.8}$$

is the conditional probability of D given a separation of the data based on the values of a and $S_a(v)$ is the subset of D where feature a takes value v. [69]

Merit is calculated by summing the ranks (1 for the feature with the highest information gain, 2 for the second, etc.) of each feature across all 10 folds, then dividing by the number of folds.

The average merit for the features are: 0.688 for the $Sensor - ID$, 0.656 for the room ($Cluster$), 0.118 for the time of day ($Time$) and 0.021 for the day of the week (Day).

Using these features, the decision tree correctly classifies 98.0% of the tracks. These results show us that (a) there is sufficient separation in the dataset to support the idea of an unsupervised learning approach, and (b) as Figure 5.16 explicitly visualizes, the location plays a significant role in the classification – a much higher than either chronological feature – whereby the merit of the day-of-the-week feature is almost negligible.

5.3.3. Unsupervised

As the decision tree shows, the tracking data is sufficient to distinguish between two residents. However, recording and labeling of training data is a long and laborious task. Therefore, we

reject the idea of a system requiring recording and labeling to work. Instead, we will study unsupervised learning methods, so that we can separate the tracks of two or more residents without such a tedious setup process.

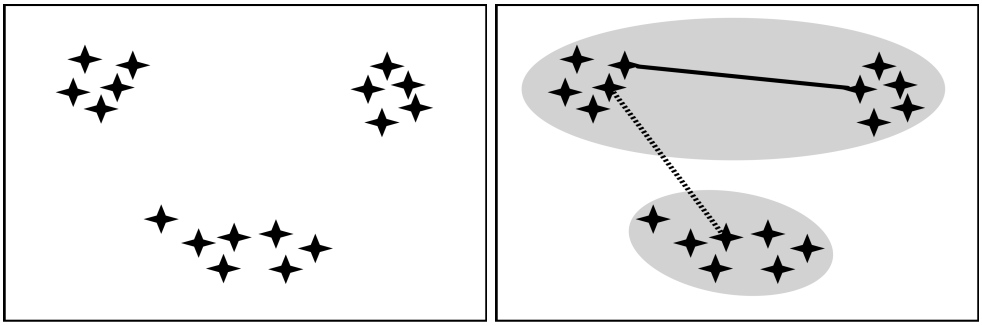
Fuzzy Clustering One of the simplest and most common clustering algorithms is *k-means* [54]. However, k-means strictly assigns all entities (in this case: tracks) to one cluster. Thus, k-means is likely too strict as we assume that there is an intra-individual variation of activity features that could cause clusters to overlap. For example, gait speed might vary depending on the time of day, or the circadian rhythm (getting up, going to sleep) might vary between weekdays. Also, k-means has no concept of noise and as such will forcefully cluster all entities, regardless of their distance to other entities.

To accommodate noise and cluster overlap, we instead apply Fuzzy C-means [26], a fuzzy or soft clustering algorithm. Instead of return a strict assignment of entities to clusters, Fuzzy C-means returns a matrix of cluster membership probabilities: the membership probabilities of all entities to all clusters. The closer an entity is to a cluster center, the higher is its membership probability.

The membership probabilities also offer a simple approach to filtering uncertain data: we can choose to exclude entities in further analysis if its largest membership probability falls under a certain threshold. For example, it seems more certain that entity e_1 with membership array $[m(c_i), m(c_j)] = [0.8, 0.2]$ belongs to cluster c_i than that e_2 with membership values $[0.4, 0.6]$ belongs to cluster c_j. We could thus elect to exclude e_2 from further analysis.

Constrained Clustering Section 5.3.2 lists the features used in the clustering and classification approach: two temporal and two location criteria. However, it is possible to derive possibly relevant meta-information from the dataset: tracks overlapping in time cannot originate from the same person, unless a tracking error occurs. While this information does not directly translate to an additional clustering feature, we can integrate it by using a constraint-based clustering algorithm and use the temporal overlap as *cannot-link* constraints. We use another extension of the k-means algorithm, COP-KMEANS clustering [92], to test the inclusion of the constraints in the process.

Unlike basic clustering algorithms, that simply aim to minimize the distance of entities to a cluster center, constrained clustering techniques also take sets of constraints that describe the relationship between data points into account. While k-means iteratively assigns entities to one of a previously defined number of clusters, then moving the cluster center to the center of its assigned data points, COP-KMEANS does so while simultaneously observing *must-link* and *cannot-link* constraints. The constraints take the form of a two-column matrix, each line describing a pair of entities that either belong (must-link) or do not belong (cannot-link) to the same cluster. Figure 5.17 shows an example of how a single must-link and cannot-link constraint can lead a clustering algorithm to a – at least visually – obscure solution. The example data (left) does not visually lend itself to separation into two clusters. K-means for $k = 2$ would either combine two of the three visual clusters into one or split one of the "visual"



(a) Unconstrained (b) Constrained

Figure 5.17.: Constrained clustering example, after Antoine et al.[4].

```
Input: Dataset D, must-link S_= ⊆ D × D, cannot-link S_≠ ⊆ D × D

Let C_1,...,C_k be the initial cluster centers.
while ¬converged
        for each point d_i ∈ D
                assign to closest cluster C_j
                such that ¬VIOLATE_CNSTRS(d_i, C_j, S_=, S_≠)
                if no such cluster exists, return
        for each cluster C_i,
                update its center by averaging over all points d_j assigned to it
return C_1,...,C_k

VIOLATE_CNSTRS(data point d, cluster C, must-link S_=, cannot-link S_≠)
        for each (d, d_=) ∈ S_=
                if(d_= ∉ C) return true
        for each (d, d_≠) ∈ S_≠
                if(d_≠ ∈ C) return true
        return false
```

Listing 5.3: Pseudocode of COP-KMEANS, after Wagstaff and Rogers [92].

clusters between the two target clusters. A must-link (continuous line) and cannot-link (dashed line) constraint helps guiding the algorithm to the correct clustering (right). Listing 5.3 shows the functioning of COP-KMEANS in pseudocode. Bold code is the “constrained” part of the algorithm. The non-highlighted lines show the basic k-means algorithm.

K-means and its constrained version suffer from two major drawbacks: first, the lack of a notion of noise means that all data points will be assigned to one of the clusters, thus affecting the results. Second, as a distance-based method it is primarily aimed at finding clusters of similar size and shape. To avoid the latter issue, we can replace k-means with a density-based clustering algorithm. One such algorithm – that also solves the problem with noise – is *DBSCAN* (Density-based spatial clustering of applications with noise) [27].

DBSCAN connects data points with many nearby neighbors to clusters. As such, it also does not expect the number of clusters as input. This seems like an advantage at first; for example, each visitor could fall into their own cluster. However, it could leave us with a large number

of clusters which then need to be associated to *another* in order to create a single personalized motion profile. Additionally, the most commonly cited constrained version of DBSCAN, *C-DBSCAN* [73], *enforces* all constraints. As such, noise in the underlying data (for example through tracking errors) that could cause noisy constraints would result in no clustering to be found at all. Due to these issues, we discard the density-based approach and continue with a constraint-based C-means approach.

Constrained Fuzzy Clustering To combine the advantages of fuzzy and constrained clustering, we turn to a constrained fuzzy clustering algorithm, Constrained Evidential C-means (CECM), as implemented by Antoine et al. [4]. Evidential clustering describes cluster membership of entities as a belief function. Just like COP-KMEANS, CECM accepts must-link and cannot-link constraints, which are then integrated into the cost function. Unlike in COP-KMEANS, however, compliance and failure of compliance of the constraints does not result in a failed clustering. Constraints may contradict each other, because compliance to constraints is a weighted parameter of the evaluation function, allowing for fine-grained control over their impact. Each constraint that is or cannot be satisfied by a cluster association reduces the value of the belief function for this association.

Equation 5.9 (after Antoine et al. [4]) shows how the compliance and violation of constraints affects the clustering process: $\mathcal{M}$ and $\mathcal{C}$ are the set of must-link and cannot-link constraints. $pl_{i\times j}(\theta)$ then describes the *plausibility* of entities x_i and x_j belonging to the same cluster, and $pl_{i\times j}(\bar{\theta})$ describes the plausibility of x_i and x_j not belonging to the same class. Equation 5.10 shows the update to the original Evidential C-means cost function $J_{ECM}(M, V)$, where M is the cluster association matrix and V describes the cluster centers. By changing θ, we can control the effect constraint compliance has on the final result.

$$J_{CONST} = \frac{1}{|\mathcal{M}| + |\mathcal{C}|}[\sum_{(x_i,x_j)\in\mathcal{M}} pl_{i\times j}(\bar{\theta}) + \sum_{(x_i,x_j)\in\mathcal{C}} pl_{i\times j}(\theta)] \tag{5.9}$$

$$J_{CECM}(M, V) = (1 - \xi)J_{ECM}(M, V) + \xi J_{CONST} \tag{5.10}$$

5.4. Simultaneous Tracking and Identification

As we will see, tracking using binary, ambient sensors works well as long as targets are separable within the data. Once two targets occupy the same or neighboring sensor areas, there is simply insufficient information to continue tracking reliably. Furthermore, since the above-mentioned identification approach relies on the locations residents occupy, tracking and identification in a household where residents spend large parts of their day or night together will be difficult.

(a) Ambient (b) Ambient & body-worn

Figure 5.18.: Discrete Bayesian networks for the ambient multi-target tracking approach and the ambient and body-worn approach

5.4.1. Approach

The most obvious solution to a lack of resolution for reliable tracking is to install more sensors or ones with a higher resolution. Looking at other application areas of multi-target tracking, we notice that, when tracks of targets overlap in time and space, there is often additional data that the tracking algorithm can refer to in order to rectify the problem. When tracking planes on a radar, for example, we can infer the velocity and acceleration of a plane based on the difference in location over two points in time. Due to inertia, a plane will not stop or turn in a matter of a few seconds. When tracking motorists and cyclists in an autonomously driving car, similar rules apply to these targets. There is no such data available for a person traversing between the sensor areas of two motion sensors.

To introduce velocity, acceleration or any other kind of motion information about a person into the tracking algorithm, we must employ other sensors. Based on the approach described below, we equip participants with smartphones and derive a basic activity model from the devices' accelerometer data. The activity model informs the MHT evaluation function. Figure 5.18 shows the discrete Bayesian networks for the original, ambient-only target state estimation and the updated ambient-plus-body-worn approach. Note that, unlike the Bayesian network of Wilson [96] as depicted in Figure 3.2a, the location information does not affect the activity model. In Wilson's work, this feature is only made possible by previously recording data and training motion models describing personal transition probabilities; in our case, the effect of the motion model on the tracking is limited to the fact that it is unlikely for a person to trigger a sensor if there is no detectable motion of that person.

5.4.2. Architecture

To incorporate the data from body-worn sensors into the system described in Section 5.2.2.3, we extend it as follows. The extended class diagram is shown in Figure 5.20[9].

[9]This diagram leaves out operations of classes that were already introduced in Figure 5.8

SensorBayesianGraphFilter The `BayesianGraphFilter` is genericized and extended by the `SensorBayesianGraphFilter`, which includes a more elaborate motion model[10]. For our evaluation, we implemented a basic (*moving – not moving*) model based on accelerometer data of a smartphone. Depending on the available data, the model can be arbitrarily complex, for example by modeling a target's speed, acceleration or current mode of movement, e.g. running, walking or sitting. Using this motion model, we can

1. associate a track with an identifier. The identifier becomes part of the hypotheses, where, as soon as an identifier appears (i.e., a smartphone registers in the system), all possible combinations of tracks and identifiers are used to generate new hypotheses. To continue the example from Section 3.1.2, a new identifier ID_1 in the system would generate nine new hypotheses (see also Figure 5.19), one for each association of an existing track with the identifier:
   - $H_2^A : \{\{T_1, ID_1\}, T_2, NT_1\}$
   - $H_2^B : \{T_1, \{T_2, ID_1\}NT_1\}$
   - $H_2^C : \{T_1, T_2, \{NT_1, ID_1\}\}$
   - $H_3^A : \{\{T_1, ID_1\}, (T_2, E_1)\}$
   - $H_3^B : \{T_1, \{(T_2, E_1), ID_1\}\}$
   - $H_4^A : \{\{(T_1, E_1), ID_1\}, T_2\}$
   - $H_4^B : \{(T_1, E_1), \{T_2, ID_1\}\}$
   - $H_5^A : \{(T_1, ID_1), T_2\}$
   - $H_5^B : \{T_1, (T_2, ID_1)\}$
2. modulate the transition probabilities of the tracking algorithm. In our case, we multiply the previous transition probability with the *moving* probability of the motion model *if* the triggered sensor is not the same as the previously triggered sensor. This limitation was implemented when it became clear that a motion sensor can be triggered with small hand, arm or head movements, which were not picked up by the accelerometer. This causes transitions between sensors to be more likely to be caused by a person in motion.

TargetState The `TargetState` class wraps the motion model data and handles time-critical operations, such as removing state information after a predefined timeout.

PathWithFilter Inside a `PathHypothesis`, `MovementPath`s can be associated with a `TargetState`. Thus, both are wrapped in a `PathWithFilter` for faster access.

[10]For a description of the motion model, see Section 3.1

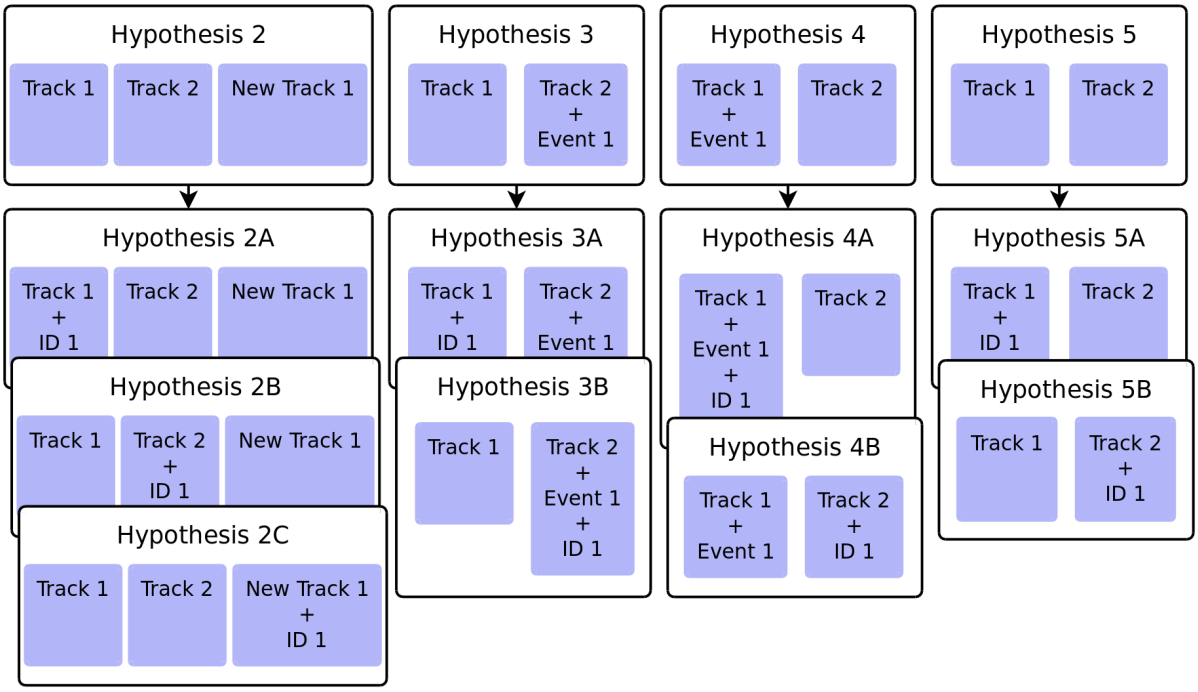


Figure 5.19.: Hypothesis generation from four hypotheses (H_2 - H_5) and an identifier (ID_1)

MovementDetection The `MovementDetection` class implements the activity state detection. It receives data from the body-worn sensors and converts them into a `TargetState`. The `IDEAALPreprocessor` is registered with the MovementDetection as a `TargetStateChangeListener` and is responsible for feeding the activity state information into the `MHT-Filter`.

5.4.3. Data Preprocessing

For this approach, any kind of device that reports motion parameters, such as of smartphones, smart watches, RFID tags or purpose-built devices such as a Shimmer sensor[11] as utilized by Marschollek et al. [59] can be used. The data contains additional information about the activity of residents, which we can *then* try and match with the information from the ambient sensors. Figure 5.21 shows an example of accelerometer data collected in our evaluation and the corresponding model output. In this case, the model only differentiates between *moving* and *not moving* based on preset acceleration and velocity thresholds.

[11] http://www.shimmersensing.com

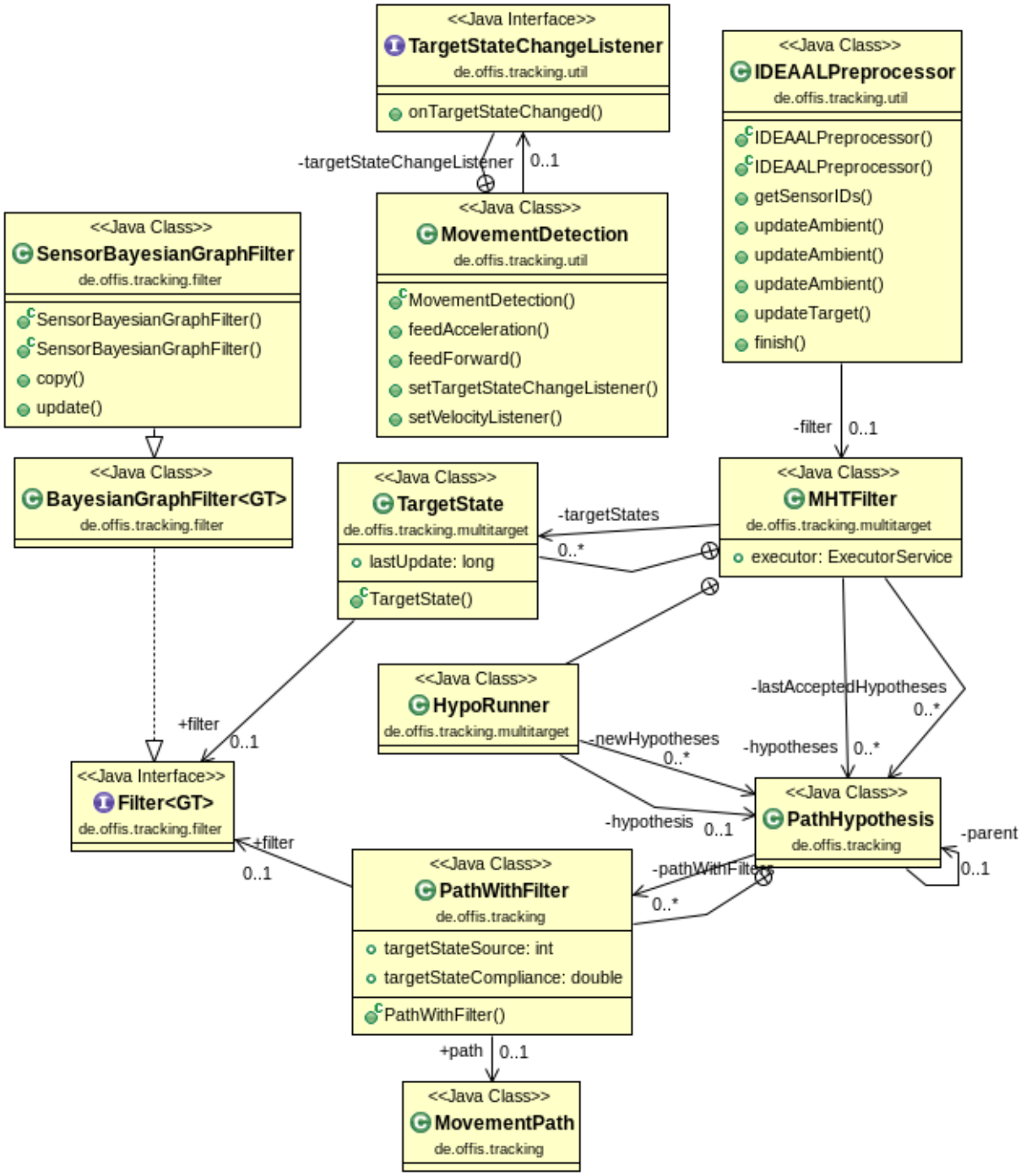


Figure 5.20.: UML class diagram of extended tracking architecture

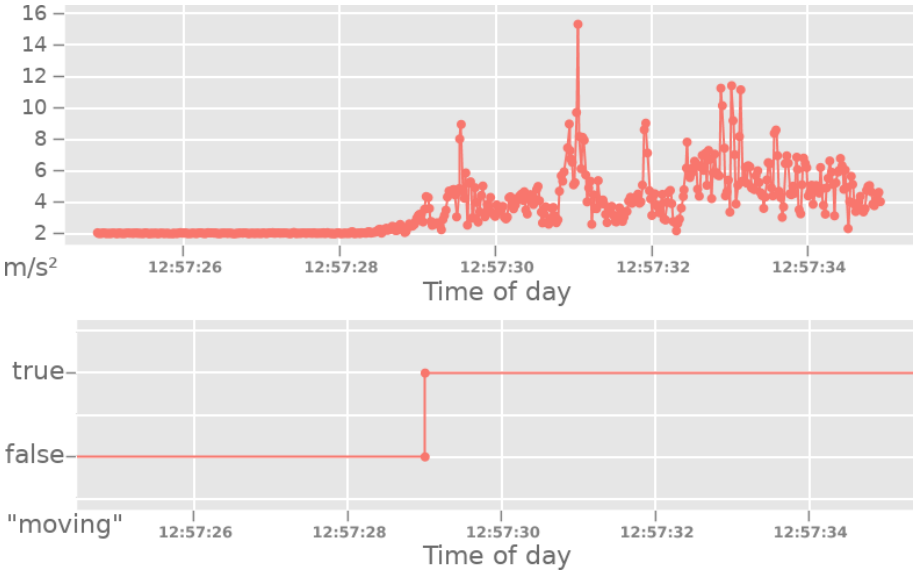


Figure 5.21.: Example output of smartphone accelerometer and motion detection model

6 Evaluation

The evaluation of this work is conducted in three phases: First, we use data from a living lab equipped with a large amount of sensors to evaluate the multi-target tracking performance around three parameters: the type of underlying graph, the window size of the algorithm and the number and placement of sensors. Second, we conduct a field trial in the home of an elderly couple. The home will be equipped with standard smart home devices in order to verify that the results from the living lab experiment hold true for real-world data. We also use the data from the first two datasets to evaluate the identification procedure via track clustering as described in Section 5.3.3. Lastly, we conduct another experiment in a living lab in order to further investigate scenarios where target tracking proved difficult and evaluate the identification approach using body-worn sensors as described in Section 5.4.

6.1. Living Lab Experiment #1

The first experiment aims to evaluate the performance of the multi-target tracking algorithm based on three main parameters: the type of underlying graph, the window size of the algorithm and the number and placement of sensors.

Generated vs. constructed graph While it may be convenient to have the underlying sensor graph generated, the graph generation might introduce errors that might be avoidable. Furthermore, not all applications allow for days or weeks of preliminary data collection.

Window size The window size of the MHT algorithm is a scaling parameter, meaning that, if the algorithm were to be run on hardware with low specifications, one would be able to reduce its computational requirements. On the other hard, if computational complexity is not an issue, one might achieve better results with a larger window size.

Sensor selection Many research projects evaluate their work in living labs, which often have a larger density of sensors than a common smart home, or in a deliberately optimized setup in field trials. In order to evaluate the performance of the tracking algorithm with varying sensor density and location, we test it on several subsets of the dataset.

The data used for this evaluation was recorded at the University of Washington's Center for Advanced Studies in Adaptive Systems (CASAS) [15]. The laboratory is a 3-bedroom, 2-story apartment and was inhabited by two volunteer students for approximately 8 months. Each bedroom has a bed, desk and closet. Bathroom, kitchen and living room are shared. The

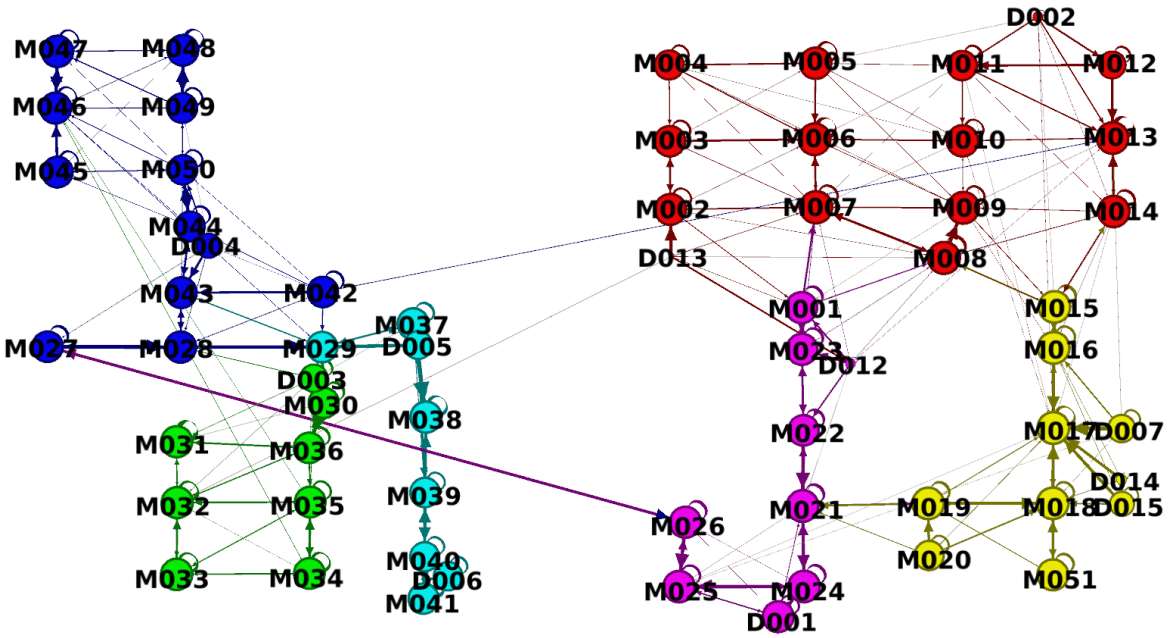


Figure 6.1.: Directed, weighted graph of a two-story living lab based on data from two residents recorded over an 8-months period. M-named nodes are motion sensors, D-named nodes are contact switches. The corresponding blueprint is shown in Figure 3.6.

laboratory is equipped with various sensor technologies, such as magnetic contact switches on doors, cabinets and refrigerator and “smart” light switches. For this evaluation, however, we consider the data recorded by the 45 motion sensors only as they already provide a very thorough coverage and precision with low amounts of noise. Each sensor is mounted on the ceiling at a height of about 2.4m and points downwards to cover an area of approximately 1.2 x 1.2m. The layout of the laboratory and the sensor placement is depicted in Figure 3.6.

6.1.1. Tracking Performance

The primary goal of the first evaluation is to see whether a sensor graph generated from prerecorded data is precise enough to allow for robust multi-target tracking. To this end, we use two months of data from our dataset to generate a sensor graph as described in Section 5.2.1.1. The graph is depicted in Figure 6.1.

Due to our specific interest in multi-target tracking, we focus on data for which both residents were present and active. The data is annotated with the identity of the resident. We chose the first 20 time frames for which the following conditions were met:

- The time frame is at least 20 minutes long or contains at least 300 sensor events,
- none of the residents remained in one room for the whole time frame, and
- neither resident is sleeping (i.e. inactive) for more than 20% of the time.

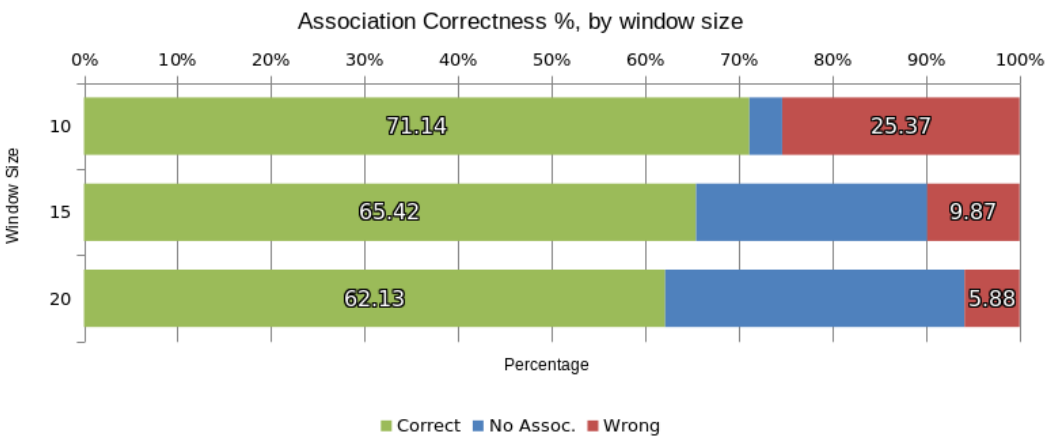


Figure 6.2.: Tracking accuracy with varying window size on data from the WSU CASAS living lab

The resulting time frames cover 6985 sensor events, where the tracks range from 10 to 338 events and last between 24 minutes and 8 hours and 50 minutes.

6.1.1.1. Generated Sensor Graph

As was described in Section 5, the approach of tracking on a graph aims to remove the necessity to add spatial information during the system setup and thereby making the installation and administration procedure unnecessarily complicated. We therefore test the tracking algorithm on two different graphs: the first graph is generated from previously recorded data from the homes where the sensors were installed (cf. Section 5.2.1.1). This approach minimizes the installation complexity but adds the requirement of prerecording data. There is also the risk of introducing erroneous edges on the graph where many subsequent sensor activations exist although no spatial neighborhood relationship exists, most notably in multi-person households.

Figure 6.2 shows that the share of wrongly associated events decreases while the share of unassociated events increases with increasing window size. While the share of wrong associations drops to 5.88% at a window size of 20, the share of unassociated events increases to 31.99%. With the smaller window size, the rate of wrongly associated sensor events is large (25.37%) while in the larger window size this rate is rather small. While it was to be assumed that the rate of wrong associations drops with a larger window size – more data results in better decisions – the share of *correctly* associated events is far lower with the larger window size due to the large share of unassociated events.

To evaluate whether this observation is based on the underlying graph or if this holds true for any graph, we run the following experiments with a manually constructed graph that models the neighborhood relations of all sensors as closely as possible.

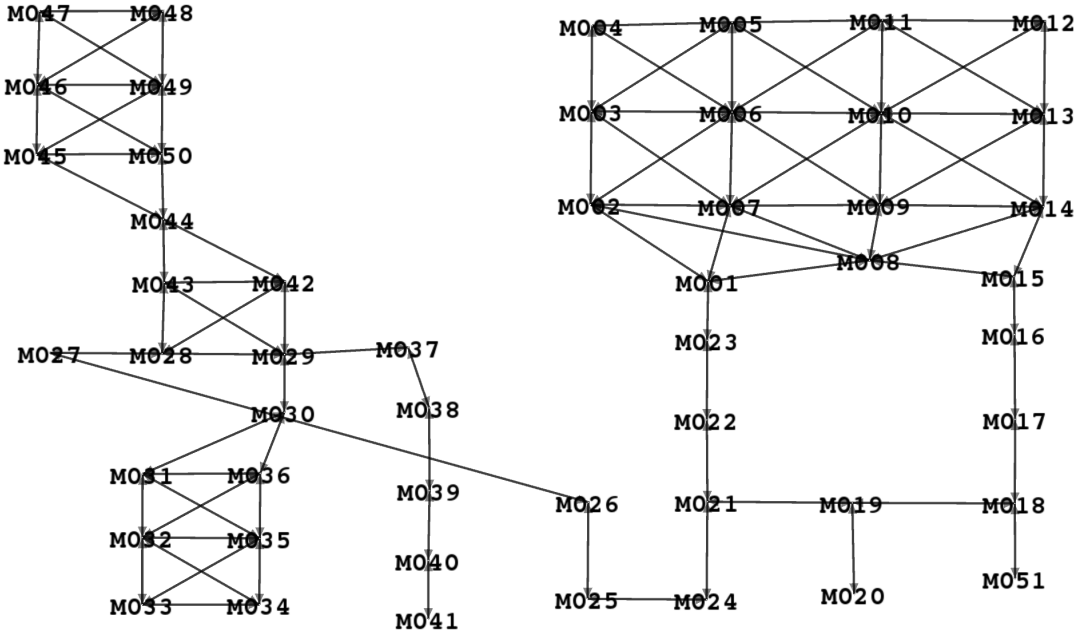


Figure 6.3.: Sensor graph for the study using data from the WSU CASAS living lab. After Crandall & Cook. [21]

6.1.1.2. Window Size

We hypothesize that the size of the window strongly influences the performance of the algorithm and that a larger window size will result in a larger number of correct associations, but also in a larger number of discarded sensor events. To test this hypothesis, we run the tracking algorithm with two window sizes: 10 and 20. We chose these values *(a)* to limit computation time, and *(b)* because the time frames we investigate during evaluation are limited in time and number of events.

Unlike the previous evaluation, this trial makes use of the constructed sensor graph in order to avoid introducing tracking errors due to erroneous graph edges. The graph is depicted in Figure 6.3. The share of correctly associated events (excluding wrong associations as well as no-association) amounts to 89.68% and 93.4% for a window size of 10 and 20, respectively (Figure 6.4). [112]

Figure 6.5 shows the performance of both runs for all time frames. While the the smaller window size results in a smaller *no-association* percentage, the larger window size results in a smaller *wrong-association* percentage and better overall performance.

As we can see, the window size of the multi-hypothesis tracking influences the association performance. The number of wrong associations in the smaller window outweighed the number of wrongly *and* unassigned events in the larger window.

Figure 6.5 shows the rate of no-association and wrong associations for all tracks in the dataset based on the tracking with a window size of 20. We can see two tracks breaking away. Looking at the track, we realize the vast majority of errors occur when only one of the targets is active. The most likely explanation of this is that two very similar hypotheses will block

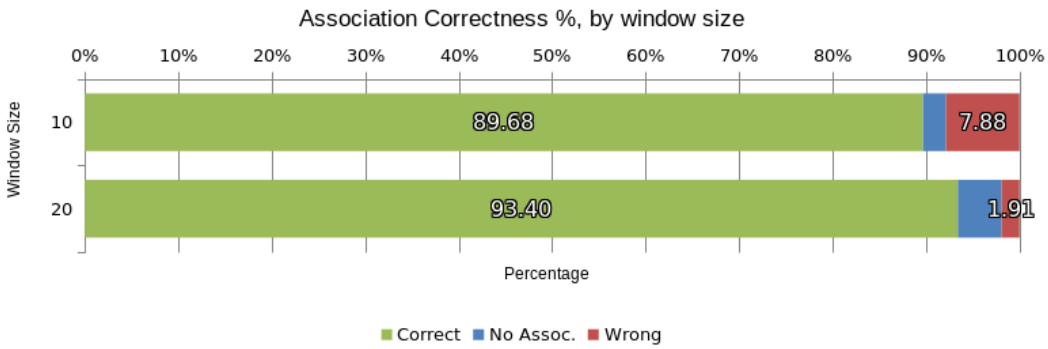


Figure 6.4.: Tracking accuracy using constructed sensor graph on data from the WSU CASAS living lab

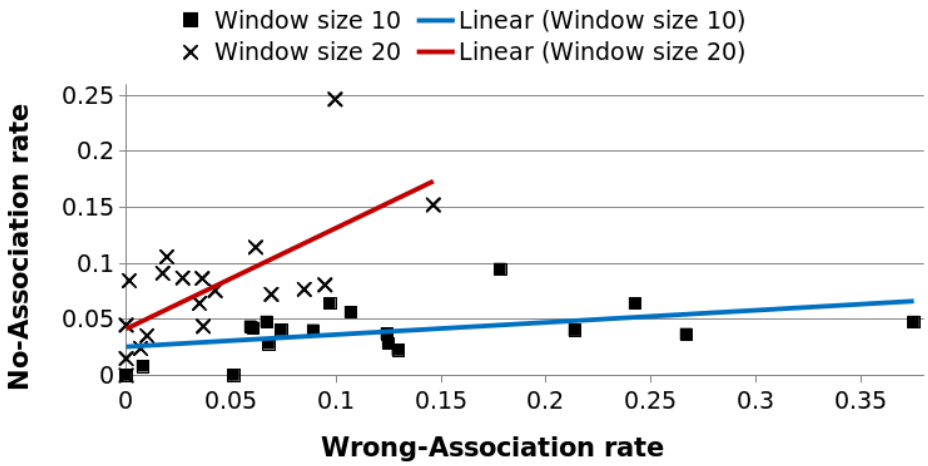


Figure 6.5.: Association error for each track, by window size

each other from being accepted, because neither one is *dominating*. Another reason could be that, due to a measurement delay, the sensor graph is too restrictive to reliably explain motion given the sensor readings.

6.1.1.3. Sensor Selection

To get a better understanding of how the number of sensors affects tracking accuracy, the algorithm is tested on subsets of the original set of sensors in decreasing size (80, 60 and 40% of the original sensor set). Instead of choosing the sensors randomly, characteristics of sensors deemed possibly influential on tracking performance were chosen:

Number of neighboring sensors Based on the assumption that sensors in doorways usually have few neighboring sensors but are important for room transition tracking, the sensors in larger areas (with many neighboring sensors) are removed. The number of neighboring sensors can be derived from the sensor graph.

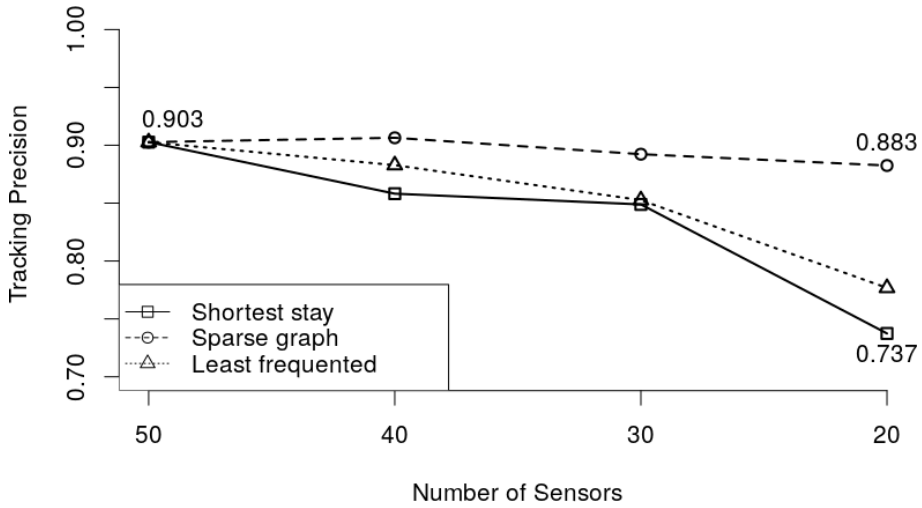


Figure 6.6.: Tracking performance across sensor groups

Duration of stay To avoid tracking a target that largely remains stationary but continues to trigger one or two sensors, we select subsets of sensors that cover areas in which the average duration of stay is short. Using previously recorded data, we can calculate he duration of stay from the duration between consecutive sensor events in the data.

Activity In order to be able to cover the largest amount of activity happening throughout the day, instead of placing sensors where tracking is simple but presence is rare, we select sensors based on the amount of activity they record. This information can be derived from previously recorded data simply by summing up the number of activations of each sensor in the dataset.

Using these criteria, we construct nine additional datasets (beside the original full dataset) and compare tracking precision for each to the original.

The resulting sensor graphs are depicted in Appendix A.7. Looking at the resulting graphs, we can see that when filtering by activity, bedrooms and the center of the living room are filtered while the hallways, bathroom and kitchen remain intact. When filtering by shortness of stay, most of the first floor (living room, kitchen) remains. Bedrooms and bathroom are filtered almost completely. When filtering by number of neighboring sensors, the hallways and narrow rooms, namely bathroom and kitchen, are filtered.

Figure 6.6 shows the tracking precision across all datasets. The results show that, the fewer sensors are used, the more important the sensor placement criteria become: while tracking precision is above 85% for all sets of 40 sensors, the precision drops significantly when there are only 20 sensors left. The precision is worst if the sensors with the shortest average duration of stay are chosen. The precision never drops below 88% when sensors with few neighboring sensors are used.

6.1.2. Identification Performance

The data for this study is taken from the results of tracking experiment #1 (Section 6.1). The approx. 7000 sensor events are combined into 112 tracks. Since the tracking algorithm occasionally produces tracks that contain data from multiple residents (referred to here as “noise”), we first test all procedures on a subset of the data which has been verified to be error-free based on the sensor data labels. Additionally, we will test the procedures also on the whole dataset, which contains tracking errors. That way, we can determine the correctness of an individual procedure and also determine its utility under realistic conditions.

6.1.2.1. Fuzzy Clustering

After preliminary tests, the *fuzzifier* was set to 2. Tests showed that higher values only stretched the results across larger filtering values, thus possibly allowing for more fine-grained control over the filtered values, but otherwise not adding any benefit.

Figure 6.7 shows that 85.9% of all tracks (at a filter value of 0.5, no filtering happens in a two-cluster scenario) are correctly clustered (Rand Index = 85.8). When filtering entities with a maximum membership value (MMV) of less than 0.6, 45% of entities are filtered. However, of the remaining data 89.7% is correctly clustered. When filtering MMVs of less than 0.7, 84.5% of data is filtered, but 100% of the remaining data is correctly clustered. [113]

As Figure 6.9 shows, clustering performance drastically decreases when the data is noisy. Without filtering, clustering is as good as a coin toss (51.8% (Rand Index 0.58)). Furthermore, no entities have an MMV of 0.6 or higher. [113]

6.1.2.2. Constrained Clustering

In the dataset without noise, COP-KMEANS produced a 91.5% precise clustering (Rand Index = 0.92). Since the algorithm does not offer a cluster membership measure like C-means, no filtering can be performed. The precision drops significantly, to 58.0% (Rand Index = 0.67), with the noisy dataset. [113]

When looking at performance under noise, we must also assume noisy constraints. Under this assumption, both COP-KMEANS and C-DBSCAN fail to find a clustering. In such a case, COP-KMEANS returns an empty set (see line 8 of Listing 5.3) while C-DBSCAN returns a cluster for each data point.

6.1.2.3. Constrained Fuzzy Clustering

When filtering MMVs below 0.4, CECM produces a 94.7% correct clustering (Rand Index: 0.95) (see Figure 6.8). Further filtering, however, merely decreases the share of correct associations. Similar to the fuzzy clustering approach, precision drops significantly (error rate of 26.0%, Rand Index: 0.71) for the noisy dataset (see Figure 6.10). [113]

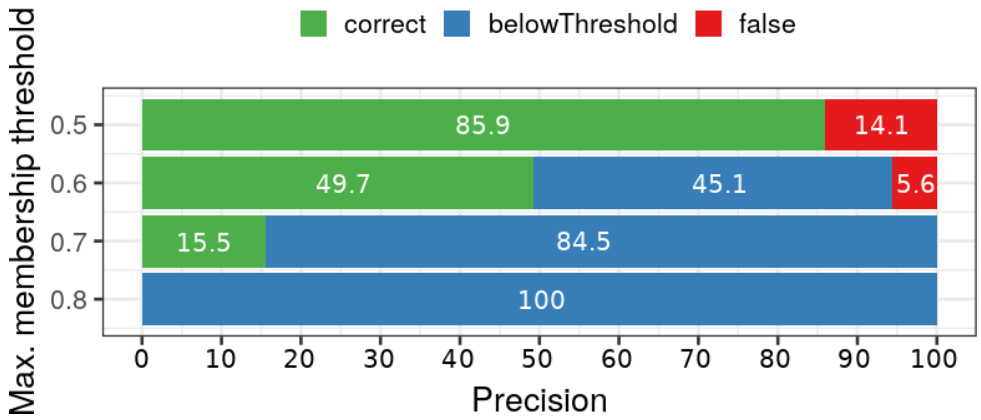


Figure 6.7.: Results of living lab tracks clustering with C-means and filtering over cluster membership grades, in %

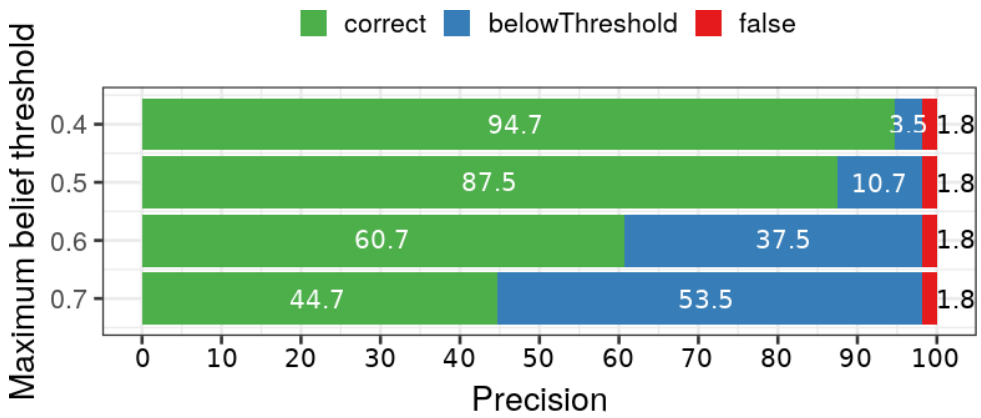


Figure 6.8.: Results of living lab tracks clustering with Constrained Evidential C-means and filtering over normalized beliefs, in %

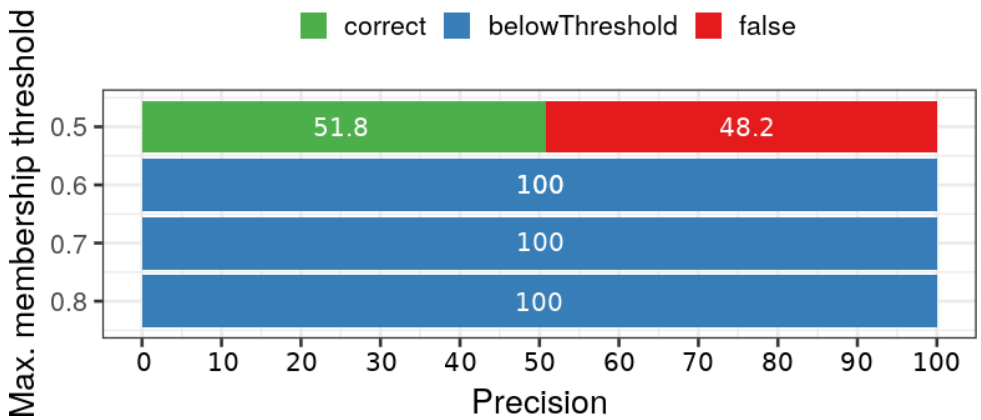


Figure 6.9.: Results of noisy living lab tracks clustering with C-means and filtering over cluster membership grades, in %

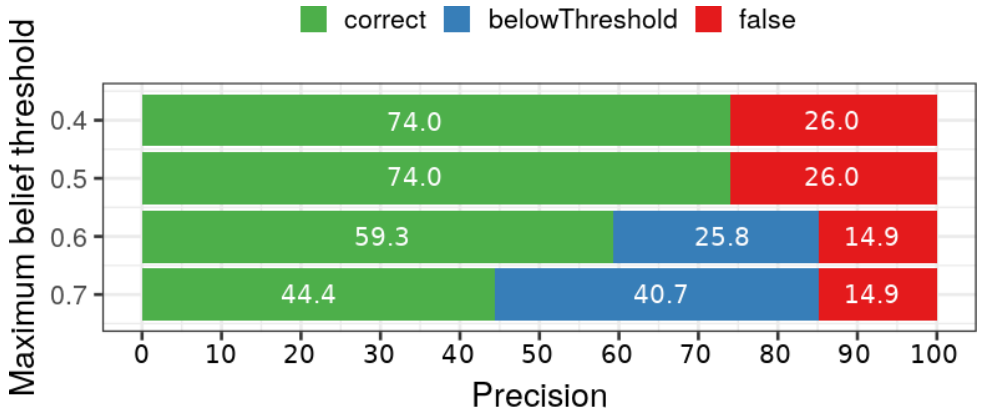


Figure 6.10.: Results of noisy living lab tracks clustering with Constrained Evidential C-means and filtering over normalized beliefs, in %

6.1.2.4. Identifying Sensor Areas

In reference to previous work [97, 61], in which a specialized identifying sensor in a central location of the home provides intermittent but accurate and precise identity data, we introduce a second set of constraints: based on preceding knowledge of residents having their own bedrooms, we associate one sensor – one of which we know that it is often triggered by one of the residents but not the other – with each resident. In this case, we associate both residents with one of the motion sensors in their respective bedroom. If certain locations in a home can be attributed to the activity of a single individual with high probability, the additional information could help improving the clustering of tracks covering these locations, which then might improve the clustering of *all* tracks. For all pairs of tracks where each track covers one of the "identifying" locations, we add a cannot-link constraint. Because CECM does not treat the constraints as axiomatic, the "identifying sensors" do not have to be triggered by only one of the residents strictly; the algorithm does not fail to converge if someone else triggers the sensor.

Adding a single identifying sensor region for each resident has a significant impact on clustering precision (see Figure 6.11). While the error rate is still significant (12.5%) when filtering MMVs up to 0.5 (Rand Index: 0.85), 87.5% of tracks are correctly assigned above that. When filtering maximum beliefs of 0.7 or less, the error disappears completely. It should be noted, however, that at this point only 17% of the original data remains. [113]

6.1.3. Summary

The goal of this first evaluation was to show the general functioning of MHT and unsupervised motion model generation and the impact the most important parameters have on its performance. In summary, the first evaluation has shown that

- using a generated sensor graph, we can correctly track two residents up to 62.13% of the time with an error rate of 5.88%.
- The constructed sensor graph is far superior to the graph generated from recorded data

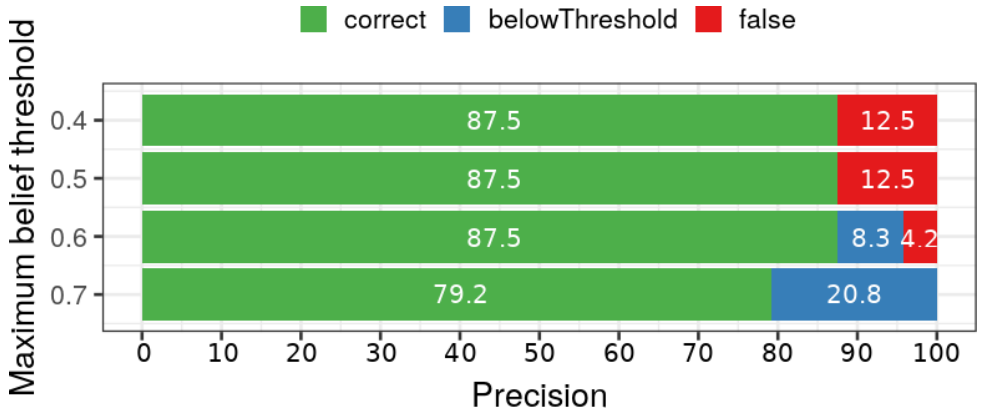


Figure 6.11.: Results of track clustering with identifying sensor areas on living lab data with Constrained Evidential C-means and filtering over maximum normalized beliefs, in %

in terms of tracking accuracy.

- The window size has a significant impact on the precision of the MHT algorithm.
- Depending on the sensor locations, tracking accuracy *might* drop significantly when the amount of sensors is reduced.
- Constrained fuzzy clustering performs best in combining motion tracks to personalized motion models among fuzzy and constrained clustering algorithms, but the error remains high.
- Using additional constraints based on locations that may identify a person, a subset of the data can be clustered with small error rates.

6.2. Field Trial

The flat equipped for the field trial is part of a retirement complex operated by the German Red Cross and inhabited by a couple (75 and 82 years) with separate bedrooms. They were recruited as part of “LivingCare”, a research project funded by the German Federal Ministry of Education and Research (grant 16SV7206). Declaration of consent, participant information and data protection notice for the study can be found in Appendices A.5 and A.6.

Figure 6.12 shows the floor plan of the flat and the locations of the sensors and actuators. The flat is approximately 80 m^2 in size. There are 38 sensors and actuators installed in the flat: ten PIR motion sensors (M, *Homematic HM-Sec-MDIR-2*), eleven contact switches on doors and windows (D, *Homematic HM-Sec-SC-2*), eleven light switches (L, *Homematic HM-PB-2-WM55-2* and *HM-LC-Sw1PBU-FM*) and six roller shutter actuators (R, *Homematic HM-LC-Bl1PBU-FM*). The data (24 hours, or 4083 events across 230 tracks, during which both residents were present at all times) were labeled by hand.

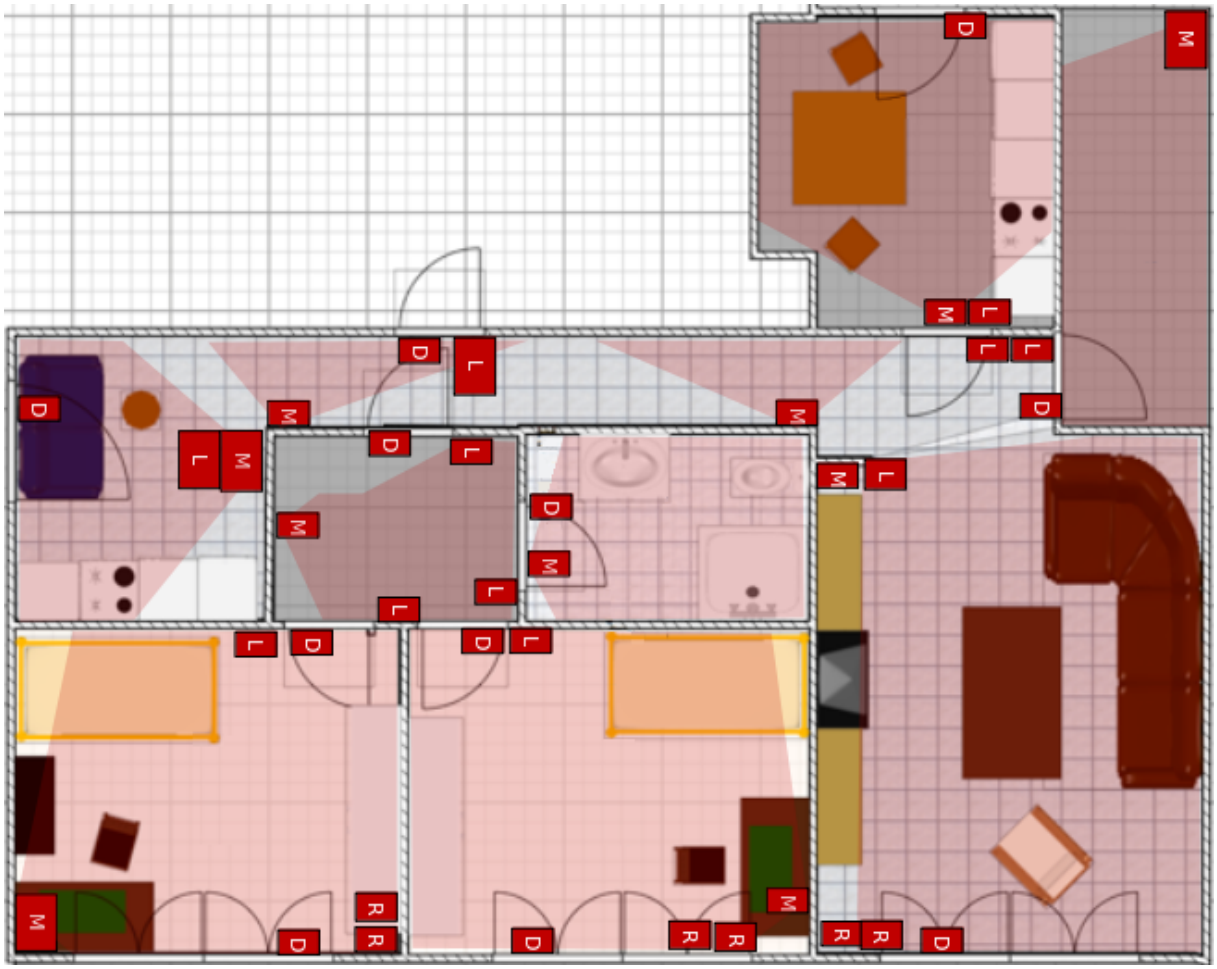


Figure 6.12.: Blueprint of the flat equipped with contact (D) and motion (M) sensors as well as light switches (L) and roller shutter (R) actuators. After Eckert et al. [102]

6.2.1. Tracking Performance

Using a window size of 10, 87.8% of the 4083 events were correctly associated. 3.5% of events were not associated, the rest was incorrectly associated. This is in line with, albeit slightly better than the results from the living lab.

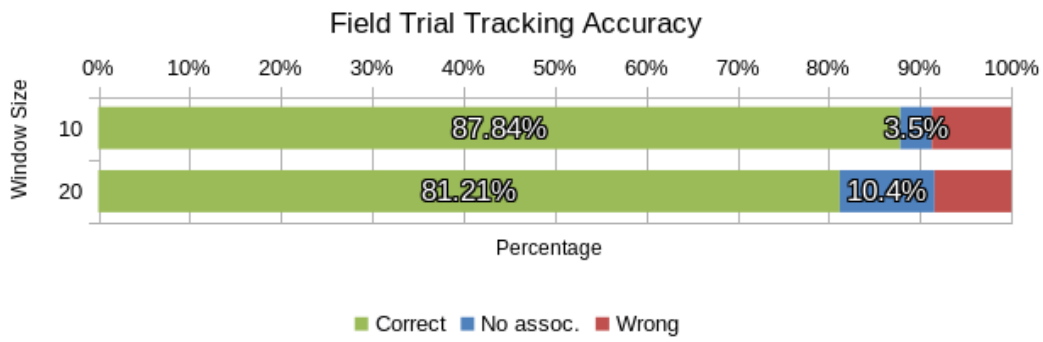


Figure 6.13.: Field trial tracking results

Surprisingly, as Figure 6.13 shows, using a window size of 20 did not improve overall performance. The rate of incorrectly associated events did not change significantly, but the rate of not associated events increased threefold, so that only 81.2% of all events was correctly associated.

6.2.2. Identification Performance

During the annotation process, it became apparent that, unlike in the first evaluation, the two residents in our trial regularly inhabited either bedroom during the day although they slept in separate bedrooms. As we saw in Section 5.3.2, location is an important factor in classification and clustering of tracks. Unless the data contains other, equally identifying data, identification will not work as efficiently here as in the previous evaluation.

As in the previous evaluation, we include constraints from temporal overlap, then test again including identifying sensors (cf. Section 6.1.2.4).

Figure 6.14 confirms the speculation from the labeling procedure: the clustering precision is significantly lower, possibly caused by location being less of an identifying feature in this household than in the previous dataset. Filtering data points with a maximum cluster membership belief of less than 0.6 removes more than 75% of all tracks. [113]

Corresponding to the previous evaluation, we also observe that an identifying sensor region for each resident has a significant impact on clustering precision (Figure 6.15): while the rate of false cluster assignments remains high for the low membership belief filters, more than one third are correctly assigned when filtering MMVs of under 0.6 (Rand Index = 0.87) with no erroneous assignments. [113]

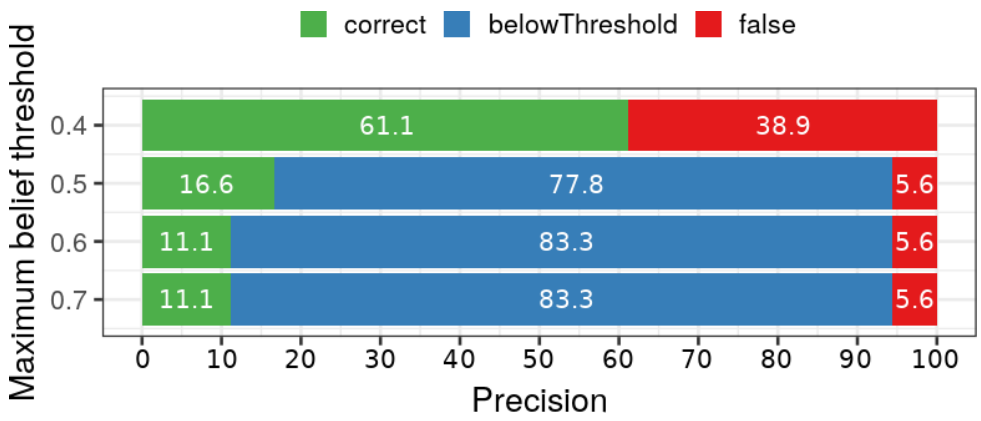


Figure 6.14.: Track clustering results in % on field trial data with CECM and filtering over normalized beliefs.

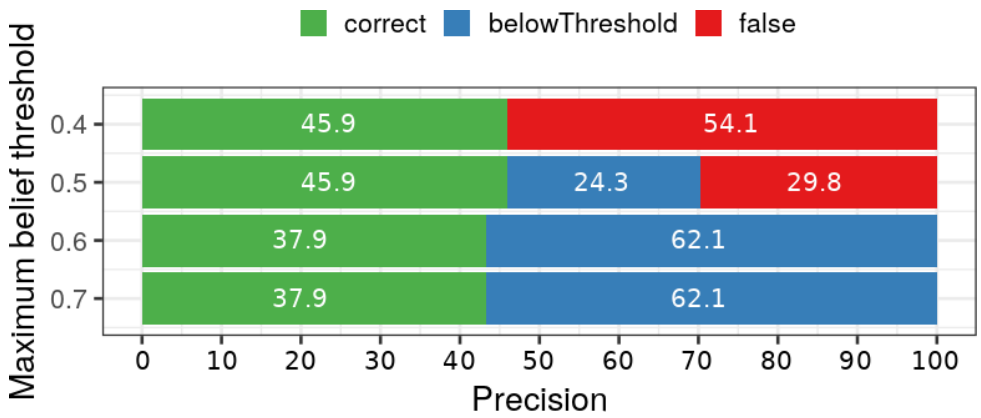


Figure 6.15.: Track clustering results in % on field trial data with CECM and filtering over normalized beliefs with identifying sensor

6.2.3. Summary

The goal of this evaluation was to show if the proposed tracking and identification approaches work as reliable under realistic conditions as they do in the laboratory. As we have seen,

- the share of correctly associated sensor events matches the findings from the laboratory.
- Increasing window size for the MHT algorithm did not improve the tracking error as it did previously, resulting in a significantly higher error rate of 9.4%.
- Unsupervised identification based on constrained fuzzy clustering does not work, unless at least some additional (e.g. location-based) constraints are added.

6.3. Living Lab Experiment #2

Since we have shown that situations in which two targets are well separable by ambient sensors can be reliably discerned by the tracking algorithm, we will here focus on situations in which the residents' tracks meet or overlap. To achieve a maximally realistic sequence of activities, we ask participants to follow an "activity script". This script aims to emulate activities of daily living, such as personal hygiene, food preparation and leisure time activities. The experiment is conducted pairwise, where both participants have a different activity script. The scripts are shown in Tables 6.1 and 6.2.

Based on the approach described in Section 5.4, we equip the participants with smartphones to derive data for a binary (`moving` – `not moving`) activity model from the devices' accelerometer. The devices are carried in the pocket, so that the sensor data is not falsified by smartphone use or hand gesturing. The model is tuned manually based on test measurements and reports a user *moving* if their speed is determined to be at or above $0.4\frac{m}{s}$ or their acceleration at or above $1\frac{m}{s^2}$.

The experiment was conducted pairwise with six participants, resulting in 3 hours of ambient sensor data and approximately 6 hours of mobile phone sensor data. On one occasion, the recording of data of one mobile phone failed to start, which was rectified 10 minutes into the experiment. The data contains 1317 ambient sensor events and approximately 1050000 mobile phone sensor readings at 50Hz. Declaration of consent, participant information and data protection notice for the study can be found in Appendices A.2, A.3 and A.4.

Figure 6.16 shows the blueprint of the living lab with ambient sensors and their sensing area drawn in. Figure 6.17 shows the sensor graph that was created for the experiment based on the blueprint. While there are more than 100 sensors and actuators installed in the lab, we limited this experiment to nine motion sensors (*Homematic HM-Sec-MDIR* and *FS20 PIRI-2-KU*), three light barriers (*FS20 IRL*) and four contact switches (*Homematic HM-Sec-SC*) to simulate a more realistic smart home setting. Based on previous experience, we omitted all KNX motion sensors except for the one in the corridor. These sensors are installed on the ceiling and usually cover the whole room. Since it is more important to be able to separate individuals with the sensors rather than covering the largest possible area, we instead rely on

Minute	Instruction	Activity
0	Sit down on the sofa in the living room	Ambulating
4	Walk up to the toilet, then return to the sofa	Elimination
7	Look into the closet in the bedroom, then sit down at the table in the kitchen	Dressing & Eating
9	Go to the bedroom and sit down on the bed	Sleeping
11	Walk into the living room to meet the other participant	Ambulating & Communicating
12	Sit down in front of the computer in the bedroom	Working / Communicating
15	Sit down on the sofa in the living room	Watching TV
17	Go to the shower	Personal hygiene
21	Sit down on the sofa in the living room	Watching TV
25	Go to the stove in the kitchen	Cooking
26	Open the cupboards in the kitchen	Cooking
27	Sit down at the kitchen table	Eating
29	Walk to the countertop	Eating
33	Sit down on the sofa in the living room	Watching TV
34	Walk to the countertop in the kitchen	Cooking
36	Sit down at the kitchen table	Cooking
41	Go to the bathroom and wash your hands	Personal hygiene
43	Sit down on the sofa in the living room	Watching TV
47	Open the front door and step outside	Ambulating / Communicating
48	Close the front door again and lie down on the bed in the bedroom	Sleeping
54	Do what you feel like	NA
59	Go to the refrigerator in the kitchen	Feeding
59.2	Help yourself to some chocolate	Feeding
59.8	You may now leave the apartment	Ambulating / Communicating

Table 6.1.: Instructions for Participant A in living lab study #2

Minute	Instruction	Activity
2	Sit down on the sofa in the living room	Ambulating
6	Sit down in front of the computer in the bedroom	Working / Communicating
9	Sit down at the kitchen table	Eating
11	Go to the living room and wait for the other participant	Ambulating & Communicating
12	Go to the stove in the kitchen. Feel free to inspect the drawers and cupboards	Cooking
15	Sit down on the sofa in the living room	Watching TV
18	Go to the bedroom and have a look into the closet	Dressing
20	Go to the stove in the kitchen	Cooking
21	Sit down on the armchair in the living room	Watching TV
27	Go to the table in the kitchen, then return to the armchair in the living room	Cooking
29	Open the front door and step outside	Ambulating
30	Close the door from the inside, then sit down on the sofa in the living room	Watching TV
33	Walk to the countertop in the kitchen	Preparing a drink
34	Sit down on the armchair in the living room	Drinking
36	Sit down at the table in the kitchen	Eating
39	Open the cupboard, then sit back down at the table	Eating
41	Walk to the sink, then back to the table	Eating
42	Go to the bathroom and wash your hands	Personal hygiene
44	Sit down on the sofa in the living room	Watching TV
47	Sit down in front of the computer in the bedroom	Working / Communicating
51	Walk up to the bed	Ambulating
52	Sit down on the sofa in the living room	Watching TV
54	Do what you feel like	NA
59	Go to the refrigerator in the kitchen	Feeding
59.2	Help yourself to some chocolate	Feeding
59.8	You may now leave the apartment	Ambulating / Communicating

Table 6.2.: Instructions for Participant B in living lab study #2

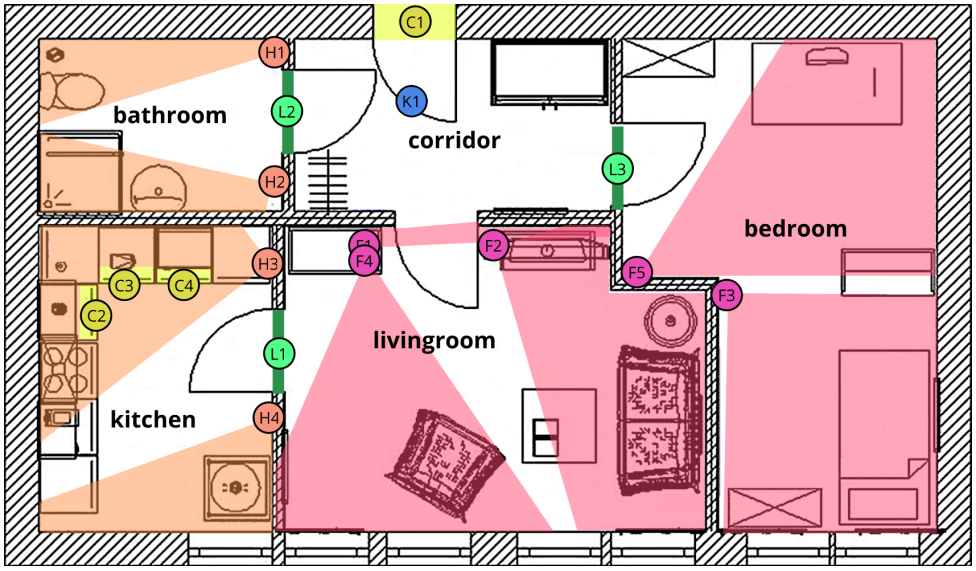


Figure 6.16.: Blueprint of the living lab equipped with light barriers (L), FS20 (F) and Homematic (H) motion sensors and contact switches (C) as well as one ceiling-mounted KNX motion sensor (K) covering the corridor

the FS20 motion sensors that can be placed more freely to cover specific areas of a room. Since the corridor doesn't have any other motion sensors, we keep the KNX sensor there.

6.3.1. Tracking Performance

In the first evaluation of the tracking algorithm's accuracy, we achieved a precision of 93.4% using the constructed sensor graph and a window size of 20. While this data was recorded in a living lab, it is still a reasonable approximation of real-world data, as the two residents lived in the lab for several months. As we have seen, the tracking precision significantly changes when both residents occupy the same or neighboring sensing areas. Therefore, this evaluation is aimed to record as many overlapping and crossing motion paths as possible. Thus, we expect the tracking accuracy on this data to be significantly lower.

The total tracking accuracy across all evaluations is 76.85%. As expected, the error rate increased compared to the first evaluation. Surprisingly, however, the share of wrongly associated events is now significantly larger relative to the share of unassociated events. While the proportion of unassociated to wrongly associated events in the first evaluation was about 2.5:1 for a window size of 20, in this evaluation it is 1:12.5 on average.

Since it cannot be expected of all residents to wear a body-worn sensors at all times, we also test how the tracking performance changes when only one of the two residents wears a body-worn sensor. Figure 6.18 shows the tracking accuracy without activity model, with one model (A and B) and with both activity models combined. As we can see, the tracking precision improves in all instances over tracking without any body-worn sensor, albeit only moderately. The overall performance is best using both activity models, with an improvement of 6.3%.

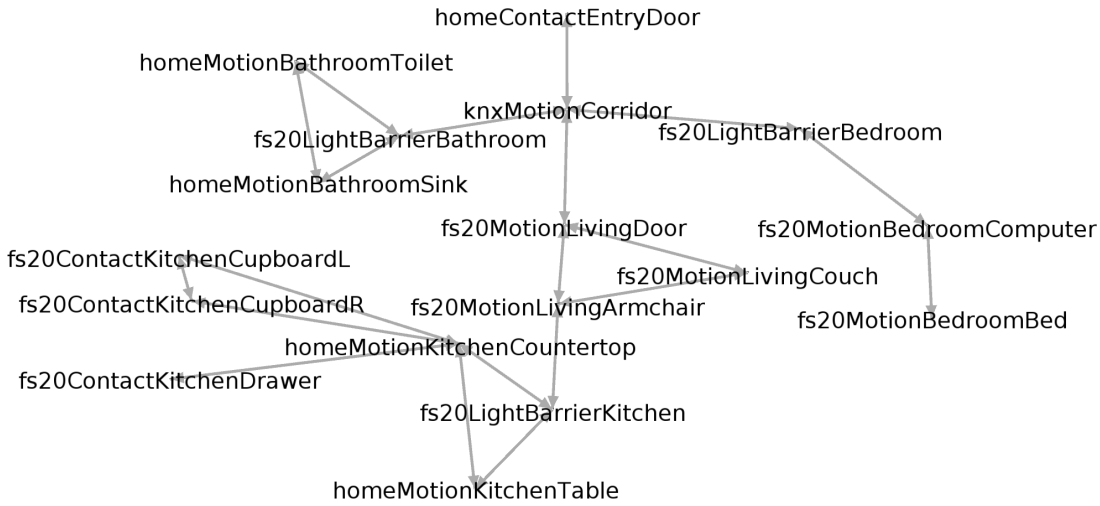


Figure 6.17.: Graph of light barriers, contact switches and FS20 motion sensors in the IDEAAL living lab for experiment #2

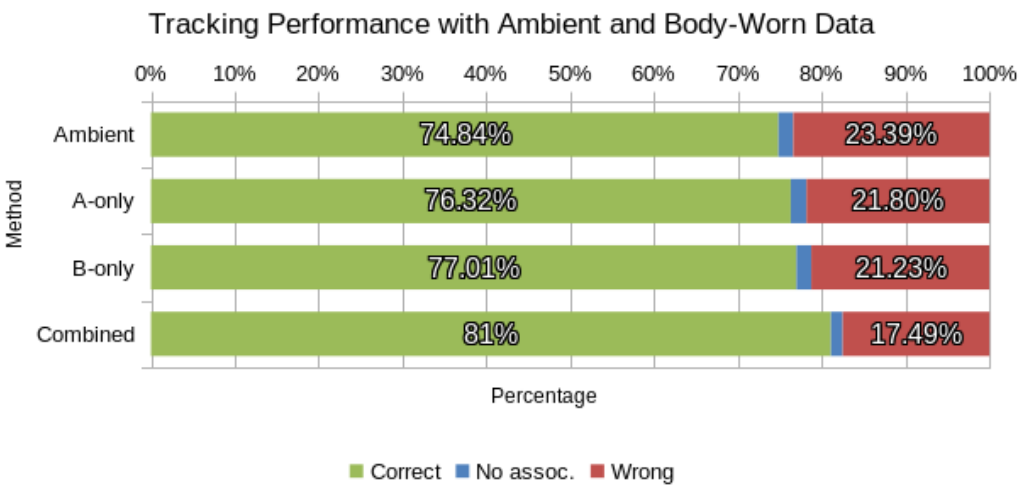


Figure 6.18.: Tracking accuracy with and without activity model in the IDEAAL living lab for experiment #2

Activity Model Output	Eval. 1	Eval. 2	Eval. 3	Average
Correct person active	49.37	45.31	55.19	**49.68**
Wrong person active	3.8	1.56	0.94	**1.92**
Both persons active	39.24	50.00	40.57	**44.09**
None active	7.59	3.13	3.3	**4.31**

Table 6.3.: Match of output of activity models to ambient sensor data, in percent

Activity Model	Eval. 1	Eval. 2	Eval. 3	Average
A	64.77	52.43	60.55	**59.0**
B	71.11	62.89	59.52	**65.08**
A+B	82.81	79.22	73.56	**78.45**

Table 6.4.: Identification accuracy using tracking with various motion models, in percent

6.3.2. Identification Performance

Before evaluating if and how body-worn sensors and an activity model improve the identification performance, we can evaluate to what extent the activity model information correlates with ambient sensor activations based on the labeled sensor event dataset. That way, we get a baseline for the identification performance in combination with the tracking algorithm. Additionally, good performance without the tracking algorithm would mean this identification approach could also be used where tracking isn't feasible, such as where no ambient sensors can be installed, computing power is minimal or a sensor graph is not available.

To do so, we calculate the activity models' output for each point in time where the activation of an ambient sensor occurred and then compare the identities of the event label and the identity of those the activity model assumes to be moving. Table 6.3 shows that, using the activity models alone, we can correctly associate *and* identify (assuming the body-worn sensor identifies the resident) an average of 49.68% of the ambient sensor events based on the fact that only one of the two persons move when the sensor is activated while – despite the simplicity of the activity model – in only 1.92% the wrong person moves.

To evaluate the identification performance using the tracking algorithm informed by the activity models, we replace the static motion model of the `BayesianGraphFilter` with a personalized motion model (`SensorBayesianGraphFilter`) which weighs the activation probabilities by the activity model derived from the accelerometer data. As we can see in Table 6.4, using an activity model for one of the two residents, 59.0 and 65.08% of events are associated to the correct identity, respectively. Using activity models for both residents, 78.45% of the track data can be correctly associated with its source.

6.3.3. Summary

The goal of this evaluation was to show to what extent body-worn sensors can be used to aid multi-target tracking and identification in situations where the resolution of ambient sensors alone is insufficient. Using a basic motion detection model based on accelerometer data, we

have found that

- compared to the ambient-only approach, the activity model containing data from both residents improved tracking accuracy by 6.28%.
- Unlike in the previous trials, the rate of false associations is high and the rate of unassociated events is low. While the former was to be expected, the reversal may hint at a mismatch of parameters.
- Using the activity model alone, we were able to match half of all ambient sensor events to body-worn sensors.
- When including activity models for both residents in the tracking algorithm, 78.45% of the ambient sensor data was correctly associated with one of the body-worn sensors.
- The error rate is higher overall compared to the previous evaluations. However, a large part can be attributed to tracking difficulties and errors.

7 Conclusion

Thanks to the growing availability of simple, affordable end user hardware, the late 20th and early 21st century has seen range of applications – from automation of heating, lighting or coffee making to the reproduction of clinical mobility assessments – emerge. As soon as the activity, which needs to be measured or detected to implement such application, extends beyond a simple sequence of one or two sensor events, basic rule-based decision making fails due to the complexity. Instead, models based mostly on supervised learning help recognize and categorize activities or similar data, such as the number of people present or the identity of a person. While supervised learning approaches mostly prove to be accurate and precise, not all applications allow for recording and labeling of training data.

In order to approach the most common cause of complexity in applications of ambient sensor applications, this work presented a multi-target tracking algorithm based on multi-hypothesis tracking and a graph of sensors and their spatial adjacency. It targets households equipped with ambient binary sensors, which are considered less obtrusive than video cameras and microphones, are easier to retrofit and usually require less maintenance. It enables the separation of data from ambient sensors generated by multiple persons in smart home environments without the need for identifying sensors. The approach makes it possible to install services of home security and ambulant care support in multi-person households, which previously have only been possible to use in single-person households or required a complex setup involving recording and labeling data.

The multi-hypothesis tracking algorithm facilitates the filtering of data that proves too complex for the sensor network resolution, so that critical applications such as mobility assessments do not cause erroneous decision making. The sensor graph is a step away from requiring a floor plan or days or week of previously recorded data of each home to be equipped for such application, and instead relies on relative localization of sensors. If possible, such a graph does not necessarily have to be constructed by hand, but can be generated from recorded data. Our first evaluation has shown, however, that a generated graph can introduce a larger error to the tracking accuracy.

While some works aim to track residents continuously, the system presented here will stop tracking a person once they become stationary for a few minutes. The main reason for this is that a low timeout of sensors allows the system to recover from tracking errors, such as when a person is assumed to be somewhere they are not. Furthermore, due to the fact that we cannot rely on personalized motion models, tracking when two people cross paths becomes near-impossible. Instead, we presented a clustering method to combine multiple tracks into

pseudonymous track clusters. Previous research on identification and separation has exclusively utilized body-worn sensors or biometric sensors and supervised learning techniques. To enable a less cost- and time-consuming procedure, we explored several clustering techniques in order to separate activity data of multiple residents in an unsupervised manner. The technique relies on the assumption that all residents are separable by features of their activity, whether based on the time of day, locations or gait speed. Basic density-based and centroid-based clustering algorithms were tested first. Fuzzy clustering procedures – similar to the hypothesis-based tracking algorithm – allow filtering of tracks which cannot be assigned to a cluster with sufficient confidence. Constraint-based clustering allows for the inclusion of inter-entity constraints to the clustering, such as the fact that two tracks cannot belong to the same cluster if they overlap temporally. Identifying sensors or information ca be added in order to create personalized motion models.

Compared to related works [10, 96, 21], the tracking algorithm works particularly well in low-resolution settings (i.e., with few binary sensors). Although discarding data for which an association to one of several persons means that some data gets lost – in our of our experiments as much as one third –, it also means that the error rate is reduced. In comparison to Wilson's [96] and Crandall's work [19], tracking accuracy of 60% does not automatically result in 40% of the remaining data associated to the wrong person. The first evaluation showed that, in a living lab, two persons can be correctly tracked up to 93.4% of the time, while the share of wrongly associated sensor events was as low as 1.2%.

It was shown that tracking accuracy varies based on the number and positioning of the sensors. While tracking of two targets in a three-room apartment using a network of 20 sensors or more can be achieved for up to 90% of the time, sensors with many neighboring sensors provide a consistently higher accuracy than those with few, and sensors in places where the duration of stay is long on average prove to be less beneficial than those where duration of stay is short. It must be noted that differences in tracking performance may not only be due to advantageous sensor placement, but also due to favorable data: While tracking in space with many adjacent sensors works well, it neglects in part space where tracking might be particularly difficult but useful, such as in narrow hallways. The share of total events covered by the different subsets of sensors in our evaluation dataset range from 11 to 98%.

To study the performance of the tracking algorithm, it was also tested in the two-bedroom apartment of an elderly couple which was equipped with a range of smart home sensors. While the error rate is significantly higher here, it was still possible to track both residents with 87.8% accuracy.

Precursory experiments using classification and supervised feature evaluation showed that, for the living lab dataset used in our experiments, location was the primary predictor for the identity of a resident. Day of week proved to be insignificant as a predictor in our datasets. Gait speed also proved difficult: first, there was great variation of gait speed in all datasets. Second, for n sensors there are, on average $3n$ pairs of adjacent sensors. Thus, for our dataset with 45 sensors, there are an additional 135 new features for gait speed alone. This increased state size requires more data to accurately measure than the few days of data we use.

The evaluation based on data from a living lab has shown that two residents can be well separated based on binary activity sensor data if they show significant differences in their whereabouts during presence. Both fuzzy and constrained fuzzy clustering successfully separated tracks of two residents based on location mostly. A fuzzy constrained clustering algorithm, *CECM*, thus proved the most successful in the task. *COP-KMEANS*, a constrained k-means algorithm proved unsuitable for the task. First, the algorithm performs a "hard" clustering, which means it does not leave room for further filtering. Second, it treats the must-link and cannot-link constraints as irrevocable: if any two constraints are contradictory, the clustering will simply fail. Although contradicting constraints might be rare in our dataset, errors introduced by the tracking algorithm might cause this procedure to fail.

The evaluation further showed that noise in the tracking data (errors in data association during the tracking procedure) significantly reduces the precision of both clustering techniques. It is likely that false associations of sensor events to residents during the tracking pre-processing causes the clustering algorithms to dismiss location as an important clustering feature, thus causing poor results.

To compensate for the noise, we finally introduced an identifying sensor area for each resident to be used as cannot-link constraints during constrained clustering. The CECM algorithm does not treat the constraints as axiomatic, but rather as an extra parameter in the cost function, whereby each violated constraint increases the cost. As a result, we are able to correctly associate 96% and 100% of two datasets of two residents after removing data with insufficient clustering confidence.

Finally, we tested the addition of data from a body-worn sensor to the tracking and identification procedure. Main reason for this approach is the widespread use of smartphones, smart watches and other personal devices equipped with activity sensors. Since it was already established that tracking is particularly difficult when two people occupy the same space or cross paths, we designed the study such that two participants will meet. We showed that, in these scenarios, the false association rate was around 20%, although the proportion of unassociated to wrongly associated events in this study suggests that there may be a parameterization issue causing too many wrong associations to be accepted.

The activity model based on accelerometer data from two participants has been shown to match nearly 50% of ambient sensor data without any tracking data. Including the activity model in the tracking algorithm's update function decreases the rate of false associations in difficult-to-track situations by about six percent and allows for almost 80% of data to be correctly associated to one of two residents. Although the error rate in our living lab evaluation was high throughout, the drop from 185 to 139 falsely associated sensor events corresponds to an error reduction of nearly 25%. Although body-worn sensor data is not always available, it was shown that the additional data helps tracking two residents, if it is available. Most notably, accuracy was improved when both residents wore accelerometers, as opposed to just one resident.

As Table 7.1 shows, the combination of adapted multi-hypothesis tracking on a graph and constrained evidential C-means clustering works better than both of the referenced previous works. In the field trial we conducted in the apartment of an elderly couple, the overall per-

Table 7.1.: Tracking and Identification performance comparison

Approach	**Setup**	**blind**	**with ident. data**
Wilson et al. [96]	Laboratory	46.0%	66.0%
Crandall & Cook [19]	Laboratory	DNA	∼60%
MHT + CECM	Laboratory	59.7%	79.3%
MHT + CECM	Field	33.3%	44.9%

formance was significantly lower. That being said, it still shows the tracking and identification approach can be used independently *or* combined to track and identify two or more people at least part of the time. The algorithm is also useful for applications of home security and technical care support, as it swiftly determines when a person has left or entered the monitored area based on fading of inactive tracks *without* requiring information about entrances and exits.

Ultimately, the tracking and identification precision required from a smart home system depends on the application. To determine the changes in gait speed of an ambulant care patient, it is sufficient to identify them once a day. To enable personalized automation, such as light switching or heating, the resident must be permanently identified. In the latter case, a biometric or body-worn solution is preferable. For the former case, we have shown a way to solve the problem without additional hardware or complicated setup. For the latter case, we have implemented a way of incorporating data from body-worn sensors into the tracking algorithm. This way, both activity data as well as identifying data is introduced. While the identification procedure is hard to compare to the state of the art because no comparable work could be found, we can see from the results of combining the tracking and identification procedures that more data can be correctly associated to one of two residents than in similar works.

8 Outlook

The results from experiments in both living labs as well as the field trial show that the multi-target tracking algorithm works similarly well across a range of sensor setups of varying number and types of sensors, though the setup in a living lab with many sensors with small sensing areas showed the best results. We have also seen that the placement of sensors can play a significant role for multi-target tracking. The next step must be to find an optimal sensor setup and derive rules about what kind of setup does and does not work. Most notably, there were few sensors with overlapping sensing areas in either experiment. The resulting rules may be a mixture of the sensor subsets and criteria examined here, and may contain criteria which were not yet considered. Furthermore, the sensor locations were chosen based on technical criteria, such as the number of neighboring sensors and duration of stay. Alternatively, it might be necessary to investigate what data is useful or necessary for a particular application, so that sensors can be installed based on other priorities. For example, when trying to recognize activities of daily living, it will be necessary to monitor kitchen, bathroom, bed and doors; preparing meals, toileting and socializing are important categories of ADLs.

The evaluation was focused on measuring tracking performance for two residents. This was chosen due to the fact that most people live in either one- or two-person households, and users of applications for care-related monitoring often live with a partner of are frequently visited. However, the approach is not theoretically limited in its applicability and studies with three or more individuals may provide better insight into the limitations of the algorithm. We assume, however, that the amount and resolution of sensors required to accurately track more than three persons quickly exceeds the acceptable burden on end users.

Another important aspect of the tracking procedure is the filtering of hypotheses. One the one hand, the hardware running the algorithm will always be a limiting factor in the number of hypotheses that can be kept in memory at one point. On the other hand, too greedy filters might remove a correct hypotheses before sufficient data is available. Finding the right parameters for each filter is a tedious, manual task at this point. It might be possible to find a correlation between the number of targets, sensors, location of sensors or sensor size and the filter parameters, so that generalizations can be drawn from individual setups to others, making the installation of the system faster and more reliable.

The identifying information used in the evaluations of the ambient-only identification procedure were simulated. While it seems reasonable to assign a motion sensor installed in a one-person bedroom to this person, we have yet to test this approach practically. Also the inclusion of an identifying sensor has not been tested practically, although Wilson [96] has conducted a

thorough evaluation involving identifying sensors.

Before using clustering to group multiple tracks of a single person into one cluster, we mentioned several tracking approaches that rely on personalized motion models for all residents or use *personalized graphs* for tracking. While we now possess the data to implement tracking based on a personalized motion model, we have not pursued this approach so far. It must be assumed that it would improve tracking, but it is unclear how significant potential errors originating from the tracking *or* clustering procedure would negatively affect the performance of the tracking using such models.

The improvement of tracking performance using ambient *and* body-worn sensors shows little proof that the burden imposed by wearing body-worn sensors might be worth accepting. While the inclusion of the body-worn sensor caused little overhead in the living lab setting, the implementation of the architecture necessary to capture a stream of accelerometer data or similar is not always possible. Lastly, the evaluation was limited to a binary activity model based on accelerometer data. Further studies could attempt to include data of other sensors, such as gyroscope or barometer to generate a more precise activity model with more than two target states.

The approach of clustering pseudonymous tracks into motion profiles suggested in this article does not solve the identification problem. Identification requires either a body-worn sensor or supervised learning. However, our results show that, at least for a subset of the data, it is sufficient to provide vague identifying information concerning a small area of the monitored space to correctly associate large amounts of data. The results using a simulated “identifying sensor” for clustering constraints should further be compared to the performance given an actual identifying sensor, whether this is a centrally located RFID reader or a biometric sensor.

Bibliography

[1] AGRAWAL, Rakesh ; SRIKANT, Ramakrishnan: Mining sequential patterns. In: *Proceedings of the Eleventh International Conference on Data Engineering* Volume 95, 1995, p. 3–14

[2] ALEMDAR, Hande ; ERSOY, Cem: Multi-resident activity tracking and recognition in smart environments. In: *Journal of Ambient Intelligence and Humanized Computing* 8 (2017), Nr. 4, p. 513–529

[3] AMERICAN HOSPITAL ASSOCIATION: *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*. `https://eur-lex.europa.eu/eli/reg/2016/679/oj`. 2005. – Accessed: 2020-01-19

[4] ANTOINE, Violaine ; QUOST, Benjamin ; MASSON, M-H ; DENOEUX, Thierry: CECM: Constrained evidential C-means algorithm. In: *Computational Statistics & Data Analysis* 56 (2012), Nr. 4, p. 894–914

[5] ARRAS, Kai O. ; CASTELLANOS, José A ; SCHILT, Martin ; SIEGWART, Roland: Feature-based multi-hypothesis localization and tracking using geometric constraints. In: *Robotics and Autonomous Systems* 44 (2003), Nr. 1, p. 41–53

[6] ARRAS, Kai O. ; GRZONKA, Slawomir ; LUBER, Matthias ; BURGARD, Wolfram: Efficient people tracking in laser range data using a multi-hypothesis leg-tracker with adaptive occlusion probabilities. In: *2008 IEEE International Conference on Robotics and Automation* IEEE (event), 2008, p. 1710–1715

[7] BARGER, Tracy S. ; BROWN, Donald E. ; ALWAN, Majd: Health-status monitoring through analysis of behavioral patterns. In: *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 35 (2005), Nr. 1, p. 22–27

[8] BARRY, Henry C. ; RICH, Brent S. ; CARLSON, R T.: How exercise can benefit older patients: A practical approach. In: *The Physician and sportsmedicine* 21 (1993), Nr. 2, p. 124–140

[9] BELL, Eric T.: Exponential polynomials. In: *Annals of Mathematics* 35 (1934), Nr. 2, p. 258–277

[10] BLACKMAN, Samuel S.: Multiple hypothesis tracking for multiple target tracking. In: *Aerospace and Electronic Systems Magazine, IEEE* 19 (2004), Nr. 1, p. 5–18

[11] BOUCHACHIA, Abdelhamid ; PEDRYCZ, Witold: Data clustering with partial supervision. In: *Data Mining and Knowledge Discovery* 12 (2006), Nr. 1, p. 47–78

[12] CHAPRON, Kévin ; BOUCHARD, Kévin ; GABOURY, Sébastien: Real-time Gait Speed Evaluation at Home. In: *Proceedings of the 5th EAI International Conference on Smart Objects and Technologies for Social Good*, 2019, p. 55–60

[13] CHOSET, Howie ; BURDICK, Joel: Sensor-based exploration: The hierarchical generalized voronoi graph. In: *The International Journal of Robotics Research* 19 (2000), Nr. 2, p. 96–125

[14] CLARKSON-SMITH, Louise ; HARTLEY, Alan A.: Relationships between physical exercise and cognitive abilities in older adults. In: *Psychology and aging* 4 (1989), Nr. 2, p. 183

[15] COOK, Diane J. ; SCHMITTER-EDGECOMBE, Maureen: Assessing the quality of activities in a smart environment. In: *Methods of information in medicine* 48 (2009), Nr. 5, p. 480–485

[16] COURTNEY, Karen L. ; DEMIRIS, George ; RANTZ, Marilyn ; SKUBIC, Marjorie: Needing smart home technologies: the perspectives of older adults in continuing care retirement communities. In: *Informatics in primary care* 16 (2008), Nr. 3, p. 195

[17] CRANDALL, Aaron S.: *Behaviometrics for Multiple Residents in a Smart Environment*, Washington State University, Ph.D. thesis, 2011. – 271 p

[18] CRANDALL, Aaron S. ; COOK, Diane J.: Coping with multiple residents in a smart environment. In: *Journal of Ambient Intelligence and Smart Environments* 1 (2009), Nr. 4, p. 323–334

[19] CRANDALL, Aaron S. ; COOK, Diane J.: Tracking Systems for Multiple Smart Home Residents. In: *Human Behavior Recognition Technologies* (2010)

[20] CRANDALL, Aaron S. ; COOK, Diane J. ; GUESGEN, Hans W. (Editor) ; MARSLAND, Stephen (Editor): *Human Behavior Recognition Technologies*. IGI Global, mar 2013. – URL `http://www.igi-global.com/chapter/tracking-systems-multiple-smart-home/75288/`. – ISBN 9781466636828

[21] CRANDALL, Aaron S. ; COOK, Diane J.: Tracking systems for multiple smart home residents. In: *Human behavior recognition technologies: Intelligent applications for monitoring and security*. IGI Global, 2013, p. 111–129

[22] CULHANE, Karen M. ; O'CONNOR, M ; LYONS, D ; LYONS, GM: Accelerometers in rehabilitation medicine for older adults. In: *Age and ageing* 34 (2005), Nr. 6, p. 556–560

[23] DAS, Sajal K. ; COOK, Diane J.: Health monitoring in an agent-based smart home by activity prediction. In: *Proceedings of the International Conference on Smart Homes and Health Telematics* Volume 14, 2004, p. 3–14

[24] DE, Debraj ; SONG, Wen-Zhan ; XU, Mingsen ; WANG, Cheng-Liang ; COOK, Diane ; HUO, Xiaoming: Findinghumo: Real-time tracking of motion trajectories from anonymous

binary sensing in smart environments. In: *2012 IEEE 32nd International Conference on Distributed Computing Systems* IEEE (event), 2012, p. 163–172

[25] DEMIRIS, George ; HENSEL, Brian K. ; SKUBIC, Marjorie ; RANTZ, Marilyn: Senior residents' perceived need of and preferences for "smart home" sensor technologies. In: *International journal of technology assessment in health care* 24 (2008), Nr. 1, p. 120–124

[26] DUNN, Joseph C.: A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. (1973)

[27] ESTER, Martin ; KRIEGEL, Hans-Peter ; SANDER, Jörg ; XU, Xiaowei et al: A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Kdd* Volume 96, 1996, p. 226–231

[28] EUROPEAN UNION: *AHA annual survey database*. `https://www.aha.org/taxonomy/term/228`. 2016. – Accessed: 2020-02-24

[29] FANG, Jian-Shuen ; HAO, Qi ; BRADY, David J. ; GUENTHER, Bob D. ; HSU, Ken Y.: Real-time human identification using a pyroelectric infrared detector array and hidden Markov models. In: *Optics express* 14 (2006), Nr. 15, p. 6643–6658

[30] FEDERAL STATISTICAL OFFICE: *Germany's population by 2050. Results of the 11th coordinated population projection*. `https://www.destatis.de/EN/Themes/Society-Environment/Population/Population-Projection/Publications/Downloads-Population-Projection/germany-population-2050.html`. 2008. – Accessed: 2020-02-22

[31] FLOECK, Martin ; LITZ, Lothar: Activity-and inactivity-based approaches to analyze an assisted living environment. In: *Emerging Security Information, Systems and Technologies, 2008. SECURWARE'08. Second International Conference on* IEEE (event), 2008, p. 311–316

[32] FORTIN-SIMARD, Dany ; BOUCHARD, Kevin ; GABOURY, Sebastien ; BOUCHARD, Bruno ; BOUZOUANE, Abdenour: Accurate passive RFID localization system for smart homes. In: *2012 IEEE 3rd International Conference on Networked Embedded Systems for Every Application (NESEA)* IEEE (event), 2012, p. 1–8

[33] FOX, Dieter ; HIGHTOWER, Jeffrey ; LIAO, Lin ; SCHULZ, Dirk ; BORRIELLO, Gaetano: Bayesian filtering for location estimation. In: *IEEE pervasive computing* (2003), Nr. 3, p. 24–33

[34] FRENKEN, Thomas ; STEEN, Enno-Edzard ; BRELL, Melina ; NEBEL, Wolfgang ; HEIN, Andreas: Motion Pattern Generation and Recognition for Mobility Assessments in Domestic Environments. In: *Proceedings of the 1st Int. Living Usability Lab Workshop on AAL Latest Solutions*, 2011, p. 3–12

[35] FRENKEN, Thomas ; VESTER, Björn ; BRELL, Melina ; HEIN, Andreas: aTUG: Fully-automated timed up and go assessment using ambient sensor technologies. In: *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2011 5th International Conference on* IEEE (event), 2011, p. 55–62

[36] GALASKO, Douglas ; BENNETT, David ; SANO, Mary ; ERNESTO, Chris ; THOMAS, Ronald ; GRUNDMAN, Michael ; FERRIS, Steven: An inventory to assess activities of daily living for clinical trials in Alzheimer's disease. In: *Alzheimer disease and associated disorders* (1997)

[37] GERDES, Verena ; SIMON, Steffen ; PAULS, Alexander ; ROLFES, H ; SCHEVE, Christine: Unterstützung der Selbstbestimmung am Lebensende: Empowerment für Patienten in der spezialisierten ambulanten Palliativversorgung (SAPV)–das BMBF-Projekt Ċicely.̇ In: *Zeitschrift für Palliativmedizin* 15 (2014), Nr. 03

[38] GIETZELT, Matthias ; WOLF, Klaus-Hendrik ; MARSCHOLLEK, Michael ; SONG, Bianying ; HAUX, Reinhold: Rekonstruktion der Topologie einer Wohnung mit Hilfe eines Installationsbus-Systems. In: *Informatik 2009–Im Focus das Leben*, Gesellschaft für Informatik e. V., 2009, p. 940–945

[39] HALL, Mark ; FRANK, Eibe ; HOLMES, Geoffrey ; PFAHRINGER, Bernhard ; REUTEMANN, Peter ; WITTEN, Ian H.: The WEKA data mining software: an update. In: *ACM SIGKDD explorations newsletter* 11 (2009), Nr. 1, p. 10–18

[40] HARLE, Robert K. ; HOPPER, Andy: The potential for location-aware power management. In: *Proceedings of the 10th international conference on Ubiquitous computing* ACM (event), 2008, p. 302–311

[41] HAYES, Tamara L. ; ABENDROTH, Francena ; ADAMI, Andre ; PAVEL, Misha ; ZITZELBERGER, Tracy A. ; KAYE, Jeffrey A.: Unobtrusive assessment of activity patterns associated with mild cognitive impairment. In: *Alzheimer's & Dementia* 4 (2008), Nr. 6, p. 395–405

[42] HIGHTOWER, Jeffrey ; BORRIELLO, Gaetano: Location systems for ubiquitous computing. In: *Computer* 34 (2001), Nr. 8, p. 57–66

[43] HOSSAIN, HM S. ; AL HAIZ KHAN, MD A. ; ROY, Nirmalya: DeActive: scaling activity recognition with active deep learning. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2 (2018), Nr. 2, p. 1–23

[44] KITBUTRAWAT, Nathavuth ; YAMAGUCHI, Hirozumi ; HIGASHINO, Teruo: Easytrack: Zero-calibration smart-home tracking system. In: *Journal of Information Processing* 27 (2019), p. 445–455

[45] KNUTH, Donald E.: *The art of computer programming*. Volume 3. Pearson Education, 1997

[46] KÖNIG, Alexandra ; CRISPIM JUNIOR, Carlos F. ; DERREUMAUX, Alexandre ; BENSADOUN, Gregory ; PETIT, Pierre-David ; BREMOND, François ; DAVID, Renaud ; VERHEY, Frans ; AALTEN, Pauline ; ROBERT, Philippe: Validation of an automatic video monitoring system for the detection of instrumental activities of daily living in dementia patients. In: *Journal of Alzheimer's Disease* 44 (2015), Nr. 2, p. 675–685

[47] KRÖGER, Tobias ; BRELL, Melina ; MÜLLER, Frerk ; LIPPRANDT, Myriam ; HELMER, Axel ; HEIN, Andreas: IDEAAL-Der Mensch im Mittelpunkt. In: *Proceedings of Deutscher AAL Kongress* Volume 4, 2011

[48] LAWTON, Mortimer P. ; BRODY, Elaine M.: Assessment of older people: self-maintaining and instrumental activities of daily living. In: *The gerontologist* 9 (1969), Nr. 3_Part_1, p. 179–186

[49] LEVENSHTEIN, Vladimir I.: Binary codes capable of correcting deletions, insertions, and reversals. In: *Soviet physics doklady* Volume 10, 1966, p. 707–710

[50] LIAO, Lin ; FOX, Dieter ; HIGHTOWER, Jeffrey ; KAUTZ, Henry ; SCHULZ, Dirk: Voronoi tracking: Location estimation using sparse and noisy sensor data. In: *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)* Volume 1 IEEE (event), 2003, p. 723–728

[51] LOGAN, Beth ; HEALEY, Jennifer ; PHILIPOSE, Matthai ; TAPIA, Emmanuel M. ; INTILLE, Stephen: A long-term evaluation of sensing modalities for activity recognition. In: *UbiComp 2007: Ubiquitous Computing*. Springer, 2007, p. 483–500

[52] LOTFI, Ahmad ; LANGENSIEPEN, Caroline ; MAHMOUD, Sawsan M. ; AKHLAGHINIA, M. J.: Smart homes for the elderly dementia sufferers: identification and prediction of abnormal behaviour. In: *Journal of ambient intelligence and humanized computing* 3 (2012). – URL `http://link.springer.com/article/10.1007/s12652-010-0043-x`

[53] LU, Jiakang ; SOOKOOR, Tamim ; SRINIVASAN, Vijay ; GAO, Ge ; HOLBEN, Brian ; STANKOVIC, John ; FIELD, Eric ; WHITEHOUSE, Kamin: The smart thermostat: using occupancy sensors to save energy in homes. In: *Proceedings of the 8th ACM conference on embedded networked sensor systems* ACM (event), 2010, p. 211–224

[54] MACQUEEN, James: Some methods for classification and analysis of multivariate observations. In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* Volume 1, 1967, p. 281–297

[55] MAENNER, Matthew J. ; SMITH, Leann E. ; HONG, Jinkuk ; MAKUCH, Renee ; GREENBERG, Jan S. ; MAILICK, Marsha R.: Evaluation of an activities of daily living scale for adolescents and adults with developmental disabilities. In: *Disability and Health Journal* 6 (2013), Nr. 1, p. 8–17

[56] MAHONEY, Florence I. ; BARTHEL, Dorothea W.: Functional evaluation: the Barthel Index: a simple index of independence useful in scoring improvement in the rehabilitation of the chronically ill. In: *Maryland state medical journal* (1965)

[57] MARINAKIS, Dimitri ; DUDEK, Gregory: Sensor Network Topology Inference. In: *Neural Information Processing Systems Workshop: Intelligence Beyond the Desktop (Whistler, Canada)*, 2005

[58] MARINAKIS, Dimitri ; DUDEK, Gregory ; FLEET, David J.: Learning sensor network topology through monte carlo expectation maximization. In: *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on* IEEE (event), 2005, p. 4581–4587

[59] MARSCHOLLEK, Michael ; BECKER, Marcus ; BAUER, Jürgen M ; BENTE, Petra ; DASENBROCK, Lena ; ELBERS, Katharina ; HEIN, Andreas ; KOLB, Gerald ; KÜNEMUND, Harald ; LAMMEL-POLCHAU, Christopher et al: Multimodal activity monitoring for home rehabilitation of geriatric fracture patients–feasibility and acceptance of sensor systems in the GAL-NATARS study. In: *Informatics for Health and Social Care* 39 (2014), Nr. 3-4, p. 262–271

[60] MICHAIL, Dimitrios ; KINABLE, Joris ; NAVEH, Barak ; SICHI, John V.: JGraphT - A Java library for graph data structures and algorithms. In: *CoRR* abs/1904.08355 (2019). – URL `http://arxiv.org/abs/1904.08355`

[61] MOKHTARI, Ghassem ; ZHANG, Qing ; NOURBAKHSH, Ghavameddin ; BALL, Stephen ; KARUNANITHI, Mohanraj: BLUESOUND: A new resident identification sensor—Using ultrasound array and BLE technology for smart home platform. In: *IEEE Sensors Journal* 17 (2017), Nr. 5, p. 1503–1512

[62] MORRIS, Meg E. ; ADAIR, Brooke ; MILLER, Kimberly ; OZANNE, Elizabeth ; HANSEN, Ralph ; PEARCE, Alan J. ; SANTAMARIA, Nick ; VIEGA, Luan ; LONG, Maureen ; SAID, Catherine M.: Smart-home technologies to assist older people to live well at home. In: *Journal of aging science* 1 (2013), Nr. 1, p. 1–9

[63] NAVARRO, Gonzalo: A guided tour to approximate string matching. In: *ACM computing surveys (CSUR)* 33 (2001), Nr. 1, p. 31–88

[64] NGUYEN, Nam ; VENKATESH, Svetha ; BUI, Hung: Recognising behaviours of multiple people with hierarchical probabilistic model and statistical data association. In: *BMVC 2006: Proceedings of the 17th British Machine Vision Conference* British Machine Vision Association (event), 2006, p. 1239–1248

[65] OGAWA, Mitsushiro ; SUZUKI, Ryoji ; OTAKE, Sakuko ; IZUTSU, Takeshi ; IWAYA, Tsutomu ; TOGAWA, Tatsuo: Long-term remote behavioral monitoring of the elderly using sensors installed in domestic houses. In: *Engineering in Medicine and Biology, 2002.*

24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society EMBS/BMES Conference, 2002. Proceedings of the Second Joint Volume 3 IEEE (event), 2002, p. 1853–1854

[66] OH, Songhwai ; SASTRY, Shankar: Tracking on a graph. In: *Proceedings of the 4th international symposium on Information processing in sensor networks* IEEE Press (event), 2005, p. 26

[67] PERRY, James T. ; KELLOG, Scott ; VAIDYA, Sundar M. ; YOUN, Jong-Hoon ; ALI, Hesham ; SHARIF, Hamid: Survey and evaluation of real-time fall detection approaches. In: *2009 6th International Symposium on High Capacity Optical Networks and Enabling Technologies (HONET)* IEEE (event), 2009, p. 158–164

[68] PODSIADLO, D. ; RICHARDSON, S.: The timed Up & Go: a test of basic functional mobility for frail elderly persons. In: *Journal of the American geriatrics Society* 39 (1991), Nr. 2, p. 142–148

[69] QUINLAN, John R.: Induction of decision trees. In: *Machine learning* 1 (1986), Nr. 1, p. 81–106

[70] QUINLAN, John R.: *C4. 5: programs for machine learning*. Elsevier, 2014

[71] REID, Donald B.: An algorithm for tracking multiple targets. In: *Automatic Control, IEEE Transactions on* 24 (1979), Nr. 6, p. 843–854

[72] ROTHGANG, Heinz ; MÜLLER, Rolf ; UNGER, Rainer ; KLIE, Thomas ; GÖHNER, Anne ; SCHUMACHER, Birgit: *Themenreport P̈flege 2030¨-Was ist zu erwarten, was ist zu tun*. `https://www.bertelsmann-stiftung.de/fileadmin/files/BSt/Publikationen/GrauePublikationen/GP_Themenreport_Pflege_2030.pdf`. 2012. – "Accessed: 2020-02-22"

[73] RUIZ, Carlos ; SPILIOPOULOU, Myra ; MENASALVAS, Ernestina: C-dbscan: Density-based clustering with constraints. In: *International workshop on rough sets, fuzzy sets, data mining, and granular-soft computing* Springer (event), 2007, p. 216–223

[74] SAVVIDES, Andreas ; SRIVASTAVA, Mani ; GIROD, Lewis ; ESTRIN, Deborah: Localization in sensor networks. In: *Wireless sensor networks*. Springer, 2004, p. 327–349

[75] SCHULZ, Dirk ; FOX, Dieter ; HIGHTOWER, Jeffrey: People tracking with anonymous and id-sensors using rao-blackwellised particle filters. In: *Proc. of the International Joint Conference on Artificial Intelligence*, 2003, p. 921–928

[76] SHIN, Jae H. ; LEE, Boreom ; PARK, Kwang S.: Detection of abnormal living patterns for elderly living alone using support vector data description. In: *Information Technology in Biomedicine, IEEE Transactions on* 15 (2011), Nr. 3, p. 438–448

[77] SHRIVASTAVA, Nisheeth ; MADHOW, Upamanyu ; SURI, Subhash: Target tracking with binary proximity sensors: fundamental limits, minimal descriptions, and algorithms. In: *Proceedings of the 4th international conference on Embedded networked sensor systems* ACM (event), 2006, p. 251–264

[78] SINGH, Jaspreet ; MADHOW, Upamanyu ; KUMAR, Rajesh ; SURI, Subhash ; CAGLEY, Richard: Tracking multiple targets using binary proximity sensors. In: *Proceedings of the 6th international conference on Information processing in sensor networks* ACM (event), 2007, p. 529–538

[79] SKUBIC, Marjorie ; GUEVARA, Rainer D. ; RANTZ, Marilyn: Testing classifiers for embedded health assessment. In: *Impact Analysis of Solutions for Chronic Disease Prevention and Management*. Springer, 2012, p. 198–205

[80] SPASOVA, Velislava ; ILIEV, Ivo: A survey on automatic fall detection in the context of ambient assisted living systems. In: *International journal of advanced computer research* 4 (2014), Nr. 1, p. 94

[81] STEEN, Enno-Edzard ; FRENKEN, Thomas ; EICHELBERG, Marco ; FRENKEN, Melina ; HEIN, Andreas: Modeling individual healthy behavior using home automation sensor data: Results from a field trial. In: *Journal of Ambient Intelligence and Smart Environments* 5 (2013), Nr. 5, p. 503–523

[82] STEEN, Enno-Edzard ; FRENKEN, Thomas ; FRENKEN, Melina ; HEIN, Andreas: Functional Assessment in Elderlies' Homes: Early Results from a Field Trial. In: *Ambient Assisted Living*. Springer, 2014, p. 3–17

[83] SULLIVAN, Josephine ; CARLSSON, Stefan: Tracking and labelling of interacting multiple targets. In: *European conference on computer vision* Springer (event), 2006, p. 619–632

[84] SULLIVAN, Josephine ; NILLIUS, Peter ; CARLSSON, Stefan: Multi-target tracking on a large scale: Experiences from football player tracking. In: *Proceedings of the IEEE ICRA 2009 workshop on people detection and tracking* Citeseer (event), 2009

[85] TAPIA, Emmanuel M. ; INTILLE, Stephen S. ; LARSON, Kent: Activity recognition in the home using simple and ubiquitous sensors. In: *International conference on pervasive computing* Springer (event), 2004, p. 158–175

[86] TEIXEIRA, THIAGO ; DUBLON, GERSHON ; SAVVIDES, ANDREAS: A survey of human-sensing: Methods for detecting presence, count, location, track, and identity. In: *ACM Computing Surveys* 5 (2010)

[87] UNITED NATIONS, DEPARTMENT OF ECONOMIC AND SOCIAL AFFAIRS, POPULATION DIVISION: *World Population Ageing 2019: Highlights*. `https://www.un.org/en/development/desa/population/publications/pdf/`

ageing/WorldPopulationAgeing2019-Highlights.pdf. 2019. – Accessed: 2020-02-22

[88] US CENSUS BUREAU: *Historical Households Tables*. https://www.census.gov/data/tables/time-series/demo/families/households.html. 2019. – Accessed: 2020-02-20

[89] U.S. ENERGY INFORMATION ADMINISTRATION: *Residential Energy Consumption Survey*. https://www.eia.gov/consumption/residential/data/2015/index.php?view=consumption. 2015. – Accessed: 2020-02-22

[90] VAUPEL, James W. ; LUNDSTROM, Hans: The future of mortality at older ages in developed countries. In: *The future population of the world: What can we assume today* (1994), p. 295–315

[91] VIRONE, Gilles ; NOURY, Norbert ; DEMONGEOT, Jacques: A system for automatic measurement of circadian activity deviations in telemedicine. In: *IEEE Transactions on Biomedical Engineering* 49 (2002), Nr. 12, p. 1463–1469

[92] WAGSTAFF, Kiri ; CARDIE, Claire ; ROGERS, Seth ; SCHRÖDL, Stefan: Constrained K-means Clustering with Background Knowledge. In: *ICML*, 2001, p. 577–584

[93] WANG, Tinghui ; COOK, Diane: sMRT: Multi-Resident Tracking in Smart Homes with Sensor Vectorization. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP (2020), 02, p. 1–1

[94] WANG, Yan ; CANG, Shuang ; YU, Hongnian: A data fusion-based hybrid sensory system for older people's daily activity and daily routine recognition. In: *IEEE Sensors Journal* 18 (2018), Nr. 16, p. 6874–6888

[95] WILLIAMS, Brie ; CHANG, Anna ; LANDEFELD, C S. ; AHALT, Cyrus ; CONANT, Rebecca ; CHEN, Helen: *Current diagnosis and treatment: geriatrics 2E*. McGraw Hill Professional, 2014

[96] WILSON, Daniel H.: *Assistive Intelligent Environments for Automatic Health Monitoring*, Carnegie Mellon University, Ph.D. thesis, 2006

[97] WILSON, Daniel H. ; ATKESON, Chris: Simultaneous tracking and activity recognition (STAR) using many anonymous, binary sensors. In: *Pervasive computing*. Springer, 2005, p. 62–79

[98] YUN, Jaeseok ; LEE, Sang-Shin: Human movement detection and identification using pyroelectric infrared sensors. In: *Sensors* 14 (2014), Nr. 5, p. 8057–8081

[99] ZENG, Yunze ; PATHAK, Parth H. ; MOHAPATRA, Prasant: WiWho: wifi-based person identification in smart spaces. In: *Proceedings of the 15th International Conference on Information Processing in Sensor Networks* IEEE Press (event), 2016, p. 4

[100] ZUO, Fei ; WITH, PHN de: Real-time embedded face recognition for smart home. In: *IEEE transactions on consumer Electronics* 51 (2005), Nr. 1, p. 183–190

Own Publications

[101] ASCHWEGE, Frerk M. von ; WORKOWSKI, Anke ; WILLEMSEN, Detlev ; MÜLLER, Sebastian M. ; HEIN, Andreas: Heart Rate Prediction for Coronary Artery Disease Patients (CAD): Results of a Clinical Pilot Study. In: *Digital Healthcare Empowering Europeans*, IOS Press, 5 2015, p. 828–832. – URL `http://ebooks.iospress.nl/volume/digital-healthcare-empowering-europeans-proceedings-of-mie2015`

[102] ECKERT, Ralf ; MÜLLER, Sebastian ; GLENDE, Sebastian ; GERKA, Alexander ; HEIN, Andreas ; WELGE, Ralph: LivingCare - An autonomously learning, human centered home automation system: Collection and preliminary analysis of a large dataset of real living situations. In: *Zukunft Lebensräume Kongress 2016 (ZL 2016)*, 2016, p. 549–557

[103] GERKA, Alexander ; LINS, Christian ; PFINGSTHORN, Max ; EICHELBERG, Marco ; MÜLLER, Sebastian ; STOLLE, Christian ; HEIN, Andreas: A Clustering-based Approach to Determine a Standardized Statistic for Daily Activities of Elderly Living Alone. In: *Proceedings of the 12th International Joint Conference on Biomedical Engineering Systems and Technologies - Volume 5: HEALTHINF,* INSTICC (event), SciTePress, 2019, p. 264–271

[104] LINS, Christian ; MÜLLER, Sebastian M. ; HEIN, Andreas: Model-based approach for posture and movement classification in working environments. In: *Ambient Assisted Living*. Springer, 2016, p. 25–33

[105] MÜLLER, Frerk ; MÜLLLER, S ; HELMER, Axel ; HEIN, Andreas: Evaluation of a generic heart rate model for exercise planning and execution across training modalities. In: *Proceedings of the 7th German AAL Conference*, 2014

[106] MÜLLER, Sebastian: Ambient activity monitoring for medical applications in multi-person households. In: *Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare*, 2014, p. 248–251

[107] MÜLLER, Sebastian ; FRENKEN, Melina ; ECKERT, Ralf ; BAUMGARTNER, Hannah ; FELSCHER, Andreas ; HEIN, Andreas: Acceptance and Usability of a Technical Support System for Ambulant Palliative Care. In: *Proceedings AAL Forum 2013*, 01 2013

[108] MÜLLER, Sebastian ; HELMER, Axel ; STEEN, Enno-Edzard ; FRENKEN, Melina ; HEIN, Andreas: Automated Clustering of Home Sensor Networks to Functional Regions for the Deduction of Presence Information for Medical Applications. In: *Wohnen–Pflege–Teilhabe–„Besser leben durch Technik “* (2014)

[109] MÜLLER, Sebastian ; STEEN, Enno-Edzard ; HEIN, Andreas: Inferring Multi-Person Presence in Home Sensor Networks. In: WICHERT, Helmut (Editor): *Ambient Assisted*

Living, Springer International Publishing, 2 2016 (Advanced Technologies and Societal Change), p. 47–56. – URL http://www.springer.com/de/book/9783319263434

[110] MÜLLER, Sebastian M. ; HEIN, Andreas: Multi-Target Data Association in Binary Sensor Networks. In: *AMBIENT 2015, The Fifth International Conference on Ambient Computing, Applications, Services and Technologies*, IARIA, jul 2015, p. 79–84. – URL http://www.thinkmind.org/index.php?view=article&articleid=ambient_2015_3_50_40044. – ISBN 978-1-61208-421-3

[111] MÜLLER, Sebastian M. ; HEIN, Andreas: Multi-Target Data Association in Binary Sensor Networks for Ambulant Care Support. In: *International Journal on Advances in Networks and Services Volume 9, Number 1 & 2, 2016* (2016)

[112] MÜLLER, S.M. ; HEIN, A.: Multi-Target Tracking in Home Sensor Networks. In: *Proc. of the Workshop on AI Problems and Approaches for Intelligent Environments*, 2014

[113] MÜLLER, Sebastian M. ; HEIN, Andreas: Tracking and Separation of Smart Home Residents through Ambient Activity Sensors. In: *Proceedings* 31 (2019), Nov, Nr. 1. – URL http://dx.doi.org/10.3390/proceedings2019031029. – ISSN 2504-3900

A
Appendix

A.1. Output of WEKA's J48 decision tree algorithm on labelled WSU data

```
Information Gain of individual features

Evaluator:    weka.attributeSelection.InfoGainAttributeEval
Search:weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1
Relation:     test-weka.filters.unsupervised.attribute.NumericToNominal-
R4-5-weka.filters.unsupervised.instance.RemoveWithValues-S0.0-Clast-Lfirst
Instances:    11490
Attributes:   5
              Date
              Time
              Sensor-ID
              Cluster
              Target-ID

Evaluation mode:10-fold cross-validation
average merit      average rank  attribute
 0.688 +- 0.002    1   +- 0      3 Sensor-ID
 0.656 +- 0.002    2   +- 0      4 Cluster
 0.118 +- 0.004    3   +- 0      2 Time
 0.021 +- 0        4   +- 0      1 Date


Classification performance

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     test-weka.filters.unsupervised.attribute.NumericToNominal-
R4-5-weka.filters.unsupervised.instance.RemoveWithValues-S0.0-Clast-Lfirst
Instances:    11490
Attributes:   5
              Date
              Time
              Sensor-ID
              Cluster
              Target-ID
Test mode:split 66.0% train, remainder test

=== Classifier model (full training set) ===

J48 pruned tree
```

```
------------------

Cluster = 0
|   Date <= 4
|   |   Time <= 55: 1 (92.0)
|   |   Time > 55
|   |   |   Time <= 511: 2 (154.0)
|   |   |   Time > 511
|   |   |   |   Time <= 541: 2 (3.0)
|   |   |   |   Time > 541: 1 (3.0)
|   Date > 4: 1 (256.0/23.0)
Cluster = 1
|   Date <= 2
|   |   Time <= 55: 1 (33.0)
|   |   Time > 55
|   |   |   Date <= 1
|   |   |   |   Time <= 201: 2 (82.0)
|   |   |   |   Time > 201: 1 (19.0)
|   |   |   Date > 1: 2 (200.0)
|   Date > 2
|   |   Time <= 469
|   |   |   Time <= 96
|   |   |   |   Time <= 72: 1 (25.0)
|   |   |   |   Time > 72: 2 (46.0/9.0)
|   |   |   Time > 96: 1 (255.0/10.0)
|   |   Time > 469
|   |   |   Time <= 541: 2 (184.0/15.0)
|   |   |   Time > 541
|   |   |   |   Date <= 5: 1 (116.0)
|   |   |   |   Date > 5
|   |   |   |   |   Time <= 1417
|   |   |   |   |   |   Time <= 1406: 1 (17.0)
|   |   |   |   |   |   Time > 1406: 2 (24.0/4.0)
|   |   |   |   |   Time > 1417: 1 (26.0)
Cluster = 2
|   Sensor-ID = M024: 1 (0.0)
|   Sensor-ID = M025: 1 (0.0)
|   Sensor-ID = M035: 1 (0.0)
|   Sensor-ID = M021: 1 (0.0)
|   Sensor-ID = M036: 1 (0.0)
|   Sensor-ID = M032: 1 (0.0)
```

```
|   Sensor-ID = M033: 1 (0.0)
|   Sensor-ID = M026: 1 (0.0)
|   Sensor-ID = M027: 1 (0.0)
|   Sensor-ID = M028
|   |   Time <= 469: 1 (58.0/13.0)
|   |   Time > 469
|   |   |   Date <= 6: 2 (23.0/4.0)
|   |   |   Date > 6: 1 (11.0/1.0)
|   Sensor-ID = M043: 1 (117.0/5.0)
|   Sensor-ID = M044: 1 (251.0/3.0)
|   Sensor-ID = M050: 1 (0.0)
|   Sensor-ID = M045: 1 (0.0)
|   Sensor-ID = M030: 1 (0.0)
|   Sensor-ID = M048: 1 (0.0)
|   Sensor-ID = M049: 1 (0.0)
|   Sensor-ID = M029
|   |   Time <= 375
|   |   |   Time <= 102
|   |   |   |   Time <= 72
|   |   |   |   |   Time <= 28
|   |   |   |   |   |   Date <= 2
|   |   |   |   |   |   |   Time <= 13: 1 (2.0)
|   |   |   |   |   |   |   Time > 13: 2 (4.0)
|   |   |   |   |   |   Date > 2: 1 (4.0/1.0)
|   |   |   |   |   Time > 28: 1 (6.0)
|   |   |   |   Time > 72: 2 (24.0/5.0)
|   |   |   Time > 102
|   |   |   |   Date <= 4: 1 (12.0)
|   |   |   |   Date > 4
|   |   |   |   |   Date <= 5
|   |   |   |   |   |   Time <= 279: 2 (3.0)
|   |   |   |   |   |   Time > 279: 1 (7.0)
|   |   |   |   |   Date > 5
|   |   |   |   |   |   Time <= 265: 1 (21.0)
|   |   |   |   |   |   Time > 265: 2 (5.0/1.0)
|   |   Time > 375: 2 (106.0/22.0)
|   Sensor-ID = M037: 1 (0.0)
|   Sensor-ID = M047: 1 (0.0)
|   Sensor-ID = M046: 1 (0.0)
|   Sensor-ID = M038: 1 (0.0)
|   Sensor-ID = M039: 1 (0.0)
```

```
|   Sensor-ID = M040: 1 (0.0)
|   Sensor-ID = M041: 1 (0.0)
|   Sensor-ID = M031: 1 (0.0)
|   Sensor-ID = M034: 1 (0.0)
|   Sensor-ID = M019: 1 (0.0)
|   Sensor-ID = M020: 1 (0.0)
|   Sensor-ID = M018: 1 (0.0)
|   Sensor-ID = M017: 1 (0.0)
|   Sensor-ID = M042: 1 (45.0)
|   Sensor-ID = M022: 1 (0.0)
|   Sensor-ID = M023: 1 (0.0)
|   Sensor-ID = M001: 1 (0.0)
|   Sensor-ID = M007: 1 (0.0)
|   Sensor-ID = M008: 1 (0.0)
|   Sensor-ID = M009: 1 (0.0)
|   Sensor-ID = M015: 1 (0.0)
|   Sensor-ID = M016: 1 (0.0)
|   Sensor-ID = M051: 1 (0.0)
|   Sensor-ID = M006: 1 (0.0)
|   Sensor-ID = M014: 1 (0.0)
|   Sensor-ID = M013: 1 (0.0)
|   Sensor-ID = M010: 1 (0.0)
|   Sensor-ID = M011: 1 (0.0)
|   Sensor-ID = M012: 1 (0.0)
Cluster = 3: 2 (3200.0/27.0)
Cluster = 4
|   Time <= 429
|   |   Date <= 2
|   |   |   Date <= 1
|   |   |   |   Time <= 185
|   |   |   |   |   Time <= 57
|   |   |   |   |   |   Time <= 28
|   |   |   |   |   |   |   Time <= 8: 1 (11.0)
|   |   |   |   |   |   |   Time > 8: 2 (27.0)
|   |   |   |   |   |   Time > 28: 1 (27.0)
|   |   |   |   |   Time > 57: 2 (13.0)
|   |   |   |   Time > 185: 1 (23.0)
|   |   |   Date > 1: 2 (49.0)
|   |   Date > 2
|   |   |   Time <= 269
|   |   |   |   Date <= 4: 1 (33.0)
```

```
|   |   |   |   Date > 4
|   |   |   |   |   Time <= 255
|   |   |   |   |   |   Date <= 5: 2 (13.0)
|   |   |   |   |   |   Date > 5: 1 (78.0/1.0)
|   |   |   |   |   Time > 255: 2 (41.0/3.0)
|   |   |   Time > 269: 1 (119.0)
|   Time > 429
|   |   Time <= 1428
|   |   |   Date <= 2
|   |   |   |   Time <= 491
|   |   |   |   |   Time <= 449: 2 (18.0)
|   |   |   |   |   Time > 449: 1 (81.0)
|   |   |   |   Time > 491: 2 (24.0)
|   |   |   Date > 2
|   |   |   |   Time <= 503
|   |   |   |   |   Time <= 435: 1 (11.0)
|   |   |   |   |   Time > 435: 2 (261.0/2.0)
|   |   |   |   Time > 503
|   |   |   |   |   Time <= 1407
|   |   |   |   |   |   Time <= 518
|   |   |   |   |   |   |   Date <= 6
|   |   |   |   |   |   |   |   Time <= 517: 2 (68.0)
|   |   |   |   |   |   |   |   Time > 517
|   |   |   |   |   |   |   |   |   Date <= 4: 1 (2.0)
|   |   |   |   |   |   |   |   |   Date > 4: 2 (4.0)
|   |   |   |   |   |   |   Date > 6
|   |   |   |   |   |   |   |   Time <= 510: 1 (20.0)
|   |   |   |   |   |   |   |   Time > 510: 2 (9.0)
|   |   |   |   |   |   Time > 518
|   |   |   |   |   |   |   Date <= 4
|   |   |   |   |   |   |   |   Time <= 534: 1 (45.0/1.0)
|   |   |   |   |   |   |   |   Time > 534
|   |   |   |   |   |   |   |   |   Time <= 541: 2 (7.0/1.0)
|   |   |   |   |   |   |   |   |   Time > 541: 1 (8.0)
|   |   |   |   |   |   |   Date > 4: 2 (7.0/1.0)
|   |   |   |   |   Time > 1407: 2 (72.0)
|   |   Time > 1428: 1 (29.0)
Cluster = 5
|   Time <= 108
|   |   Date <= 5: 1 (839.0)
|   |   Date > 5
```

```
|   |   |   Time <= 74: 2 (20.0/1.0)
|   |   |   Time > 74: 1 (366.0/10.0)
|   Time > 108: 1 (3731.0/7.0)

Number of Leaves  :  108

Size of the tree :  166


Time taken to build model: 0.04 seconds

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances        3828               97.978  %
Incorrectly Classified Instances        79                2.022  %
Kappa statistic                          0.9583
Mean absolute error                      0.0205
Root mean squared error                  0.105
Relative absolute error                  6.3515 %
Root relative squared error             26.042  %
Total Number of Instances             3907

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Cla
                 0         0          0         0         0          ?        -
1
                 0.991     0.035      0.975     0.991     0.983      0.994    1
                 0.965     0.009      0.987     0.965     0.976      0.994    2
Weighted Avg.    0.98      0.024      0.98      0.98      0.98       0.994

=== Confusion Matrix ===

    a    b    c   <-- classified as
    0    0    0 |    a = -1
    0 2249   21 |    b = 1
    0   58 1579 |    c = 2
```

A.2. Declaration of consent for study on body-worn sensors

OFFIS e.V. – Institut für Informatik

Frerk Müller-von Aschwege

Ansprechpartner für eventuelle Rückfragen:

Linus Barth

Telefon: *0177 9221249*

E-Mail: linus.barth@uol.de

Einwilligungserklärung

OFFIS e.V.– Institut für Informatik, Bereich Gesundheit

Titel der Studie: ***Evaluation eines Mehrzielverfolgungssystems basierend auf ambienten und körpernahen Sensoren***

Ich (Name des Teilnehmers /der Teilnehmerin in Blockschrift)

_____

bin **mündlich** von Herrn/Frau _____ darüber informiert worden, dass im Rahmen der Studie **Videoaufnahmen** gemacht werden und **personenbezogene Daten** erhoben und verarbeitet werden.

Im Rahmen der Studie werde ich nach mündlichen Anweisungen Bewegungsabläufe in einer Wohnung (dem IDEAAL-Wohnlabor) ausführen (zum Beispiel das Betreten eines Zimmers, das Sitzen auf einem Sitzplatz oder das Vorbeigehen an einem anderen Probanden).

Während der gesamten Zeit muss ich ein Smartphone in der Hand halten. Dieses Smartphone wird mir vor Beginn des Experiments ausgehändigt. Während des Experiments muss und sollte das Smartphone nicht von mir bedient werden. Weiter werden Videoaufnahmen von mir mit den in der IDEAL-Wohnung befindlichen Kameras erstellt. Hierbei dienen die Videoaufnahmen als Referenz der parallel aufgenommenen Sensordaten. Die Videoaufnahmen sind notwendig, um im Nachhinein prüfen zu können, an welcher Position ich mich genau befand, als zum Beispiel ein Bewegungsmelder auslöste. Dazu werden die im Rahmen der Studie erhobenen Sensordaten anhand der Videoaufnahmen beschriftet und daraufhin gelöscht. Es verbleiben nach der Löschung keine Daten, die mich identifizieren könnten. Es verbleiben lediglich anonyme Daten. Die Videoaufnahmen werden also nur kurzfristig gespeichert. Durch die von mir gemachten Videoaufnahmen bin ich jedoch bis zur Löschung der Videoaufnahmen potenziell individualisierbar, so dass es sich bei den erhobenen Daten bis dahin um personenbezogene Daten handelt.

Die Videoaufnahmen können nur unter sehr großem Aufwand vollständig anonymisiert werden. Diese Anonymisierung kann im Rahmen dieser Studie nicht gewährleistet werden. Daher besteht die sehr geringe Wahrscheinlichkeit, dass eine an der Datenauswertung beteiligte Person mich in den von mir gemachten Aufnahmen erkennt. Aus diesem Grund unterliegen alle an der Auswertung beteiligten Personen einer

absoluten Schweigepflicht und dürfen unter keinen Umständen vertrauliche Informationen an Dritte weitergeben.

Die Videoaufnahmen bzw. Daten mit Personenbezug werden auf einem passwortgeschützten Computer verschlüsselt aufbewahrt und nach der Auswertung der Daten spätestens am 31. Juli 2019 gelöscht. Die Daten werden nur zum Zeitpunkt der Auswertung entschlüsselt und danach wieder verschlüsselt. Während der Auswertung ist der dazu verwendete Computer vom Internet getrennt.

Da ich in den von mir gemachten Aufnahmen potenziell erkannt werden kann, habe ich das Recht meine Einwilligung zur Verwendung und Speicherung meiner personenbezogenen Daten - insbesondere von mir gefertigter Bild- und Tonaufnahmen und verknüpfter Sensordaten - bis zu deren Löschung bzw. deren vollständiger Anonymisierung jederzeit zu widerrufen, ohne dass mir hieraus Nachteile entstehen.

Mit dem Widerruf meiner Einwilligung stellt der OFFIS e.V. die Datenverarbeitung sofort ein, ohne dass dies die Rechtmäßigkeit der bisher erfolgten Verarbeitung berührt. Meinen Widerruf richte ich möglichst an: Sebastian Müller, Wissenschaftlicher Mitarbeiter OFFIS – Gesundheit, Escherweg 2, 26121 Oldenburg, sebastian.mueller@offis.de.

Die Einverständniserklärung für die Erstellung von Videoaufnahmen und Verarbeitung personenbezogener Daten ist freiwillig. Im Falle einer Ablehnung oder eines Widerrufs der Einwilligung entstehen für mich keinerlei Kosten oder anderweitige Nachteile. Eine Teilnahme an der Studie ist dann allerdings nicht möglich.

Ich hatte genügend Zeit für eine Entscheidung und erkläre mich hiermit bereit, dass eine Videoaufnahme von mir gemacht wird und meine personenbezogenen Daten, wie vorstehend beschrieben zu Studienzwecken, verarbeitet werden.

Eine Ausfertigung dieser Einwilligungserklärung habe ich erhalten. Ferner habe ich die beigefügten Datenschutzhinweise und die allgemeinen Informationen für Teilnehmer zur Kenntnis genommen.

Ort, Datum & Unterschrift des Teilnehmers: Name des Teilnehmers in Druckschrift:

_____ _____

Ort, Datum & Unterschrift des Versuchsleiters: Name des Versuchsleiters in Druckschrift:

_____ _____

A.3. Participant information for study on body-worn sensors

OFFIS e.V. – Institut für Informatik

Frerk Müller-von Aschwege

Ansprechpartner für eventuelle Rückfragen:

Linus Barth

Telefon: *0177 9221249*

E-Mail: linus.barth@uol.de

Allgemeine Information für Teilnehmende

OFFIS e.V.– Institut für Informatik, Bereich Gesundheit

Titel der Studie: ***Evaluation eines Mehrzielverfolgungssystems basierend auf ambienten und körpernahen Sensoren***

Herzlich willkommen bei unserer Studie zur "Evaluation eines Mehrzielverfolgungssystems basierend auf ambienten und körpernahen Sensoren"! Wir danken Ihnen für Ihr Interesse an dieser Studie.

Wir untersuchen mit dieser Studie, ob die Sensordaten eines nah am Körper getragenen Smartphones die Genauigkeit eines bestehenden Mehrzielverfolgungssystems verbessern kann.

Ablauf der Studie

Das folgende Experiment dauert insgesamt 1 Stunde.

Ihre Aufgabe ist es, nach mündlichen Anweisungen Bewegungsabläufe in einer Wohnung (dem IDEAAL-Wohnlabor) auszuführen. Dabei handelt es sich um Anweisungen wie zum Beispiel das Betreten eines Zimmers, das Sitzen auf einem Sitzplatz oder das Vorbeigehen an einem anderen Probanden.

Während der gesamten Zeit müssen Sie ein Smartphone in der Hand halten. Dieses Smartphone wird Ihnen vor Beginn des Experiments ausgehändigt. Während des Experiments muss und sollte das Smartphone nicht von Ihnen bedient werden.

Es werden keine Kontaktdaten von Ihnen aufgenommen. Es werden jedoch Videoaufnahmen mit den in der Wohnung befindlichen Kameras gemacht. Diese Videoaufnahmen werden anschließend verarbeitet und daraufhin (spätestens am 31. Juli 2019) gelöscht. Es verbleiben nur anonyme Daten.

Sollten Sie noch Fragen haben, wenden Sie sich damit bitte an den Versuchsleiter.

Freiwilligkeit und Anonymität

Die Teilnahme an der Studie ist freiwillig. Sie können jederzeit und ohne Angabe von Gründen die Teilnahme an dieser Studie beenden, ohne dass Ihnen daraus Nachteile entstehen.

Die im Rahmen dieser Studie erhobenen, oben beschriebenen Daten und persönlichen Mitteilungen werden vertraulich behandelt. So unterliegen diejenigen Projektmitarbeiter, die durch direkten Kontakt mit Ihnen über personenbezogene Daten verfügen, der Schweigepflicht. Des Weiteren wird die Veröffentlichung der Ergebnisse der Studie in anonymisierter Form erfolgen, d. h. ohne dass Ihre Daten Ihrer Person zugeordnet werden können.

Datenschutz

Informationen zum Datenschutz sind unseren anliegenden ausführlichen Datenschutzhinweisen zu entnehmen.

Aufbewahrungsfrist für die anonymisierten Daten

Die Aufbewahrungsfrist für die vollständig anonymisierten Daten beträgt mindestens 10 Jahre nach Datenauswertung, bzw. mindestens 10 Jahre nach Erscheinen einer Publikation zu dieser Studie.

Vergütung

Die Teilnahme an dieser Studie wird nicht vergütet.

A.4. Data protection notice for study on body-worn sensors

Datenschutzhinweise nach Art. 13 DSGVO

Verantwortlicher:

Für die Datenerhebung und Verarbeitung ist der OFFIS e.V. verantwortlich.

Datenverarbeitung:

Im Rahmen der Studie: „Evaluation eines Mehrzielverfolgungssystems basierend auf ambienten und körpernahen Sensoren“ erstellen wir Videoaufnahmen mit den in der IDEAAL-Wohnung befindlichen Kameras und erheben Sensordaten.

So werden Sie im Rahmen der Studie nach mündlichen Anweisungen Bewegungsabläufe in einer Wohnung (dem IDEAAL-Wohnlabor) ausführen (zum Beispiel das Betreten eines Zimmers, das Sitzen auf einem Sitzplatz oder das Vorbeigehen an einem anderen Probanden). Während der gesamten Zeit müssen Sie ein Smartphone in der Hand halten. Dieses Smartphone wird Ihnen vor Beginn des Experiments ausgehändigt. Während des Experiments muss und sollte das Smartphone nicht von Ihnen bedient werden.

Die erstellten Videoaufnahmen dienen als Referenz der im Rahmen der Studie erhobenen Sensordaten. Die Videoaufnahmen sind notwendig, um im Nachhinein prüfen zu können, an welcher Position Sie sich genau befanden, als zum Beispiel ein Bewegungsmelder auslöste. Dazu werden die im Rahmen der Studie erhobenen Sensordaten anhand der Videoaufnahmen beschriftet und daraufhin gelöscht. Es verbleiben nach der Löschung keine Daten, die Sie identifizieren könnten. Es verbleiben lediglich anonyme Daten. Die Videoaufnahmen werden also nur kurzfristig gespeichert. Auf diesen Aufnahmen sind Sie bis zur Löschung dieser potenziell individualisierbar, so dass die erhobenen Daten bis dahin personenbezogener Natur sind.

Rechtsgrundlage für die Datenverarbeitung bildet daher jeweilig Ihre freiwillig erteilte Einwilligung. Soweit besondere Kategorien personenbezogener Daten (z.B. Ihre Gesundheitsdaten) verarbeitet werden, holen wir Ihre Einwilligung nach Art. 9 Abs. 2 lit. a i.V.m. Art. 7 DSGVO ein. Erstellen wir Statistiken mit diesen Datenkategorien, erfolgt dies auf Grundlage von Art. 9 Abs. 2 lit. j DSGVO i.V.m. § 27 BDSG. Im Übrigen bildet Art. 6 lit. a DSGVO die Rechtsgrundlage für die Datenverarbeitung.

Ihre Einwilligung kann jederzeit widerrufen werden, ohne dass dies die Rechtmäßigkeit der bisher erfolgten Verarbeitung berührt. Wenn die Einwilligung widerrufen wird, stellen wir die entsprechende Datenverarbeitung ein. Ihren Widerruf richten Sie bitte möglichst an: *Sebastian Müller, Wissenschaftlicher Mitarbeiter OFFIS – Gesundheit, Escherweg 2, 26121 Oldenburg, sebastian.mueller@offis.de*.

Erhalten Sie für Ihre Studienteilnahme eine Vergütung, so haben Sie uns den Empfang der Vergütung unter Angabe Ihres Namens und Ihrer Adresse zu quittieren oder uns Ihre Kontoverbindung zwecks Überweisung der Vergütung mitzuteilen. Die hier erhobenen Daten werden separat zu den Untersuchungsdaten von uns gespeichert und verarbeitet. Die Verarbeitung und Speicherung erfolgt hier in Erfüllung einer rechtlichen Verpflichtung gemäß Art. 6 lit. c DSGVO.

Datenempfänger / Datenverwendung :

Wir beachten den Grundsatz der zweckgebundenen Datenverwendung und erheben, verarbeiten und speichern Ihre personenbezogenen Daten nur für die Zwecke, für welche Sie uns diese mitgeteilt haben. Eine Weitergabe Ihrer persönlichen Daten an Dritte erfolgt ohne Ihre ausdrückliche Einwilligung nicht.

Allerdings können Ihre Daten von uns an externe Dienstleister, z. B. IT-Dienstleister, Unternehmen, die Daten vernichten oder archivieren, Druckdienstleister, weitergegeben werden, welche uns bei der Datenverarbeitung im Rahmen einer Auftragsverarbeitung streng weisungsgebunden unterstützen. Auch die Übermittlung an auskunftsberechtigte staatliche Institution und Behörden erfolgt nur im Rahmen der gesetzlichen Auskunftspflichten oder wenn wir durch eine gerichtliche Entscheidung zur Auskunft verpflichtet werden. Eine Datenverarbeitung außerhalb der EU bzw. des EWR findet nicht statt.

Wir werden Ihre personenbezogenen Daten weder an Dritte verkaufen noch anderweitig vermarkten.

Datenlöschung:

Eine Löschung der gespeicherten personenbezogenen Daten erfolgt, wenn Sie Ihre Einwilligung zur Speicherung widerrufen, der jeweilige Speicherzweck entfallen ist oder wenn deren Speicherung aus sonstigen gesetzlichen Gründen unzulässig ist. Im Fall des Bestehens gesetzlicher Aufbewahrungsfristen werden die betroffenen Daten für die Dauer dieser Fristen archiviert.

Vorliegend werden wir die erstellten Videoaufnahmen unmittelbar nach der Verarbeitung und Auswertung, welche spätestens bis zum 31.07.2019 erfolgt, löschen. Die Daten sind danach vollständig anonymisiert.

In vollständig anonymisierter Form werden Ihre Studiendaten zudem zu Forschungszwecken mindestens zehn Jahre nach Datenauswertung, bzw. mindestens zehn Jahre nach Erscheinen einer Publikation zur jeweiligen Studie, aufbewahrt.

Rechte der betroffenen Person:

Betroffene Personen haben das Recht auf Auskunft seitens des Verantwortlichen über die sie betreffenden personenbezogenen Daten sowie auf Berichtigung unrichtiger Daten oder auf Löschung, sofern einer der in Art. 17 DSGVO genannten Gründe vorliegt, z.B. wenn die Daten für die verfolgten Zwecke nicht mehr benötigt werden. Es besteht zudem das Recht auf Einschränkung der Verarbeitung, wenn eine der in Art. 18 DSGVO genannten Voraussetzungen vorliegt und in den Fällen des Art. 20 DSGVO das Recht auf Datenübertragbarkeit.

Jede betroffene Person hat das Recht auf Beschwerde bei einer Aufsichtsbehörde, wenn sie der Ansicht ist, dass die Verarbeitung der sie betreffenden Daten gegen datenschutzrechtliche Bestimmungen verstößt. Das Beschwerderecht kann insbesondere bei einer Aufsichtsbehörde in dem Mitgliedstaat des Aufenthaltsorts der betroffenen Person oder des Orts des mutmaßlichen Verstoßes geltend gemacht werden. In Niedersachsen ist die zuständige Aufsichtsbehörde die Landesbeauftragte für den Datenschutz Niedersachsen, Barbara Thiel, Postfach 2 21, 30002 Hannover.

Kontaktdaten des Datenschutzbeauftragten:

Dr. Uwe Schläger

datenschutz nord GmbH

Konsul-Smidt-Straße 88

28217 Bremen

E-Mail: office@datenschutz-nord.de

A.5. Data protection notice and declaration of consent for field trial

Erläuterung zum Datenschutz

Während der Laufzeit der LivingCare Studie wird eine große Menge von Daten aufgezeichnet. Daten, die viele Details Ihres Lebens beinhalten. Aus den Daten ist ersichtlich, wann Sie zuhause sind, wann Sie schlafen gehen, das Bad benutzen und viele weitere Informationen. Diese Daten dürfen nicht in falsche Hände gelange. Dafür ist in der Studie gesorgt. ihre Daten werden nur verschlüsselt gespeichert und dürfen nur von ausgewählten Personen, die weiter unten genannten werden, eingesehen werden. Damit Sie jederzeit wissen, was mit ihren Daten geschieht und welche Rechte Sie haben, wurde diese Einwilligungserklärung erstellt. Bitte lesen Sie jeden Punkt sorgfältig durch und unterschreiben Sie nur, wenn Sie sicher sind verstanden zu haben, was mit ihren Daten geschieht. Bei Fragen wenden Sie sich jeder Zeit an die im Begleitblatt angegebenen Mitarbeiter. Dort wird Ihnen gerne Auskunft gegeben. Fühlen Sie sich nicht genötigt zu unterschreiben. Wenn Sie mit einem Punkt nicht einverstanden sind, versuchen wir das in ihrem Interesse zu klären.

Die Datenverarbeitung durch den OFFIS e. V. erfolgt nach den Vorgaben der einschlägigen Datenschutzgesetze und auf Basis der ULD Studie und ist von der datenschutz nord GmbH überprüft und akzeptiert worden.

Zweckbindung

In LivingCare wird ein System entwickelt, das ihr Verhalten untersucht. Auf Grundlage dieser Untersuchungen wird ein Computerprogramm errechnen, wann es z.B. am sinnvollsten ist ihre Heizung zu steuern, oder Lichter zu schalten. Die Steuerbefehle werden an eine Hausautomation weitergeleitet. Eine Hausautomation besteht aus vielen kleinen elektronischen Geräten, die verschiedene Aufgaben im Haushalt übernehmen können, in dem sie ohne Ihr dazu tun, wie eben schon genannt, Lichter einschalten oder ausschalten können, ihre Heizung regeln und so weiter. Damit diese Steuerungen auch sinnvoll sind, muss dieses System erst möglichst viel über Sie lernen. Dafür speichert das System alle Informationen ab, die über die einzelnen Geräte der Hausautomation aufgezeichnet werden können. Ohne diesen Schritt kann das System nicht funktionieren. Wenn das System trotzdem nicht wie erwartet funktionieren sollte, also zum Beispiel geht ein Licht immer zu einer falschen Zeit an, dann muss ein wissenschaftlicher Mitarbeiter die Daten anschauen und versuchen zu verstehen, warum das System falsch reagiert hat. Nur zu diesen Zwecken werden in ihre Daten in der LivingCare Studie gespeichert.

Ich habe verstanden für welche Zwecke meine Daten in der LivingCare Studie gespeichert werden und willige ein, dass meine Daten zu diesem Zweck aufgezeichnet werden dürfen.

Daten

Die in der Studie aufgezeichneten Daten werden nur durch die einzelnen Geräte der Hausautomation erzeugt. Dabei handelt es sich um folgende Komponenten und dazugehörige Daten:

- Bewegungsmelder: Ein Gerät das Bewegung erkennt und diese Bewegung auch meldet. Ist in einem Raum, den Sie betreten, ein Bewegungsmelder angebracht, erkennt das System, wann Sie den Raum betreten haben und wie lange Sie sich in dem Raum aufgehalten haben. Sind in mehreren Räumen Bewegungsmelder angebracht, kann damit die Zeit errechnet werden, wie lange Sie von einem Raum in den anderen benötigen. Ist zum Beispiel in der Küche und im Wohnzimmer jeweils ein Bewegungsmelder montiert, kann dadurch die Zeit gemessen werden, die Sie für diesen Weg benötigen.
- Lichtschalter: Ein Lichtschalter, wie Sie ihn gewohnt sind. Drücken Sie darauf geht das Licht an, drücken Sie erneut, geht es aus. Der Unterschied, das System weiß jetzt, wann Sie das Licht benutzen und speichert diese Information.
- Stromsensoren: An einige elektronische Geräte, wie ihren Fernseher, ihr Bügeleisen, ihren Herd, werden Stromsensoren angebracht. Diese Sensoren erkennen wann ein Gerät eingeschaltet ist und wann es ausgeschaltet ist. Diese Information wird gespeichert. So können später zu lange Einschaltzeiten, von zum Beispiel einem Bügeleisen, erkannt und Brände vermieden werden.
- Tür-/Fensterkontakte: Anhand dieser Sensoren kann erkannt werden, wann Sie eine Tür geöffnet oder geschlossen haben. Diese Information wird gespeichert. Damit können z.B. offen gebliebene Fenster erkannt werden und dem entsprechen die Heizung reguliert werden. Wenn die Außentür zulange offen steht kann Sie das System darüber informieren.
- Heizungsthermostate: Mit diesen Sensoren wird erkannt, wann und wie hoch sie die Heizung einstellen. Diese Information wird gespeichert. So ist es möglich herauszufinden, wann Sie einen warmen Raum benötigen und wann nicht. Das kann helfen Heizkosten zu sparen.
- Rollladenschalter: Rollladenschalter, so Sie Rollläden haben, erkennen, wann Sie den Raum verdunkeln, z.B., weil es durch direkte Sonneneinstrahlung zu warm im Raum wird. Diese Information wird gespeichert.
- Bettsensor: Dieser Sensor erkennt, wann Sie das Bett benutzen und wann Sie es verlassen. Diese Information wird gespeichert. So kann erkannt werden, wenn Sie z.B. nachts nach einem Sturz nicht in ihr Bett zurückkehren und es kann Hilfe gerufen werden.
- Lichtschranken: Mit Lichtschranken ist feststellbar, wie viele Menschen sich in einem Raum aufhalten und welche Richtung sich Personen bewegen. Diese Information wird gespeichert. So ist sichergestellt, dass das System weiß, wann jemand zu Hause ist und wann die Wohnung leer ist. Dies kann z.B. für die Aktivierung einer Alarmanlage Verwendung finden.

Das sind die Daten, die im Laufe der Studie über Sie gespeichert werden. Keine weiteren Daten werden erhoben oder gespeichert.

Verarbeitung

Die Daten werden vor Ort in den Gerätschaften verarbeitet. Die Daten verlassen zur Verarbeitung nicht das Haus. Die entstehenden Daten werden in einem ersten Verarbeitungsschritt gespeichert. Zu verschiedenen Zeiten werden kleine Programme gestartet, die mit den Daten Berechnungen ausführen. Diese Berechnungen haben direkten Einfluss darauf, wie die in ihrem Haus / ihrer Wohnung verbaute Hausautomation reagiert, also wann z.B. Heizungen eingeschaltet werden oder Rollos geschlossen werden. Zudem werden aus den Daten auch automatisch Rückschlüsse gezogen, ob Ihnen eventuell ein Unfall passiert sein könnte und von Ihnen befugte Personen werden durch eine Nachricht informiert. In einem regelmäßig wiederholten Arbeitsablauf löscht das System automatisch nicht mehr benötigte Daten.

Speicherung

Die Daten sollen durch den unerlaubten Zugriff dritter geschützt werden. Aus dem Grund werden die anfallenden Daten nur in der geschützten Datenbank, die auf einem Gerät bei Ihnen zuhause installiert ist, gespeichert. Eine dezentrale, also nicht bei Ihnen zuhause durchgeführte, Speicherung ist nicht vorgesehen. Die Daten werden nach Beendigung der Studie anonymisiert, d.h. es werden alle Informationen die Ihre Person und die Daten in Verbindung bringen können gelöscht. Die Daten werden zu Forschungs- und Lehrzwecken weitergenutzt werden.

Ich habe verstanden wie die Daten in der LivingCare Studie gespeichert werden und willige ein, dass die Daten zu diesem Zweck verwendet werden dürfen.

Zugriffe

Auf die Daten dürfen nur ausgesuchte Personen zugreifen. Ein Zugriff ist auch über das Internet möglich damit eine Wartung aus der Ferne des Systems möglich ist. Der Zugriff wird durch die Eingabe von Benutzername und Passwort geschützt. Der Zugriff aus dem Internet ist zusätzlich verschlüsselt. Auf Ihre Daten zugreifen dürfen in diesem Projekt nur zwei Mitarbeiter: Ralf Eckert vom OFFIS - Institut für Informatik und Sebastian Müller von der Leuphana Universität Lüneburg. Beide Mitarbeiter sind nicht berechtigt die Daten an Dritte weiterzugeben. Im Rahmen eines Mitarbeiterwechsels werden Sie über den Wechsel der Zuständigkeiten informiert. Sollte aus Wartungsgründen der Zugriff auf das System auch von anderen Mitarbeitern erforderlich sein, werden Sie darüber informiert, allerdings erhalten diese Mitarbeiter keinen Zugriff auf die Daten, sondern können nur Einstellungen am System verändern, wie z.B. wenn die Heizung immer zur falschen Zeit eingeschaltet wird, können diese Mitarbeiter solch eine Einstellung ändern. Auch bei diesen Personen wird es sich im Rahmen des Projekts nur um Mitglieder der Forschungsgruppe handeln. Es wird niemand ohne ihr Wissen auf die Daten zugreifen. Alle Zugriffe werden überwacht und festgehalten (geloggt).

Risiken

Trotz aller Vorsichtsmaßnahmen besteht immer ein minimales Restrisiko. Im Falle eines Wohnungseinbruchs können die Daten gestohlen werden und in falsche Hände gelangen. Durch die passwortgeschützte Speicherung ist ein ungewollter Zugriff auf die Daten jedoch sehr unwahrscheinlich. Das Gleiche gilt für das Internet. Selbst wenn es einem unbefugtem Dritten gelingen sollte auf die Daten zuzugreifen kann dieser Dritte ohne Hintergrundwissen nichts mit den Daten anfangen, da die Informationen ohne ihren Wohnkontext gespeichert werden. Das Risiko durch Datenmissbrauch ist daher als sehr gering einzuschätzen.

Ich habe verstanden welche Risiken in Bezug auf meine Daten in der LivingCare Studie bestehen und willige ein, dass die Daten zu diesem Zweck verwendet werden dürfen.

Rechte

Sie haben in Bezug auf Ihre Daten einige Rechte die wir Ihnen hier aufzeigen möchten. So können Sie:

- Jederzeit Einsicht auf die Daten verlangen
- Verlangen, dass Ihnen jemand genau erklärt was diese Daten bedeuten
- Der Speicherung wiedersprechen
- Das System für eine Zeitlang selbstständig deaktivieren
- Die Löschung Ihrer Daten verlangen

Zudem haben Sie das Recht jederzeit die Teilnahme an diesem Projekt zu beenden, ohne dass dadurch Nachteile für Sie entstehen. In dem Fall bauen wir die verbauten Gerätschaften wieder aus, beseitigen etwaige Beschädigung an Wänden und Tapeten.

Freiwilligkeit

Ich bestätige mit meiner Unterschrift, alle vorangehenden Angaben gelesen und verstanden zu haben. Ich leiste diese Unterschrift freiwillig und ohne Druck von Außen.

_____ _____

Datum, Ort Unterschrift

Einwilligung in Bild- oder Videoaufnahmen

Sofern Sie dem zustimmen, werden von den vorgenannten Versuchen zu Auswertungszwecken Bild- oder Videoaufnahmen angefertigt.

Hiermit willige ich ein, dass im Rahmen des Versuchs Bild- oder Videoaufzeichnungen von mir angefertigt werden.

☐ ja ☐ nein

Falls Sie die vorstehende Frage mit „ja“ beantwortet wurde:

Sofern Bild- oder Videoaufnahmen für Präsentation oder Publikationen verwendet werden, sind die aufgenommenen Personen derart unkenntlich zu machen, dass eine Wiedererkennung ausgeschlossen ist.

Hiermit willige ich ein, dass die im Rahmen des Versuchs angefertigten Bild- oder Videoaufzeichnungen für wissenschaftliche Präsentationen und Publikationen in anonymisierter Form öffentlich gezeigt werden.

☐ ja ☐ nein

Die Einwilligung kann jederzeit, ohne das Nachteile für Sie entstehen, wiederrufen werden.

_____ _____

Datum, Ort Unterschrift

Wir bedanken uns im Voraus für Ihre Bereitschaft zur Mitwirkung an unserer Arbeit!

A.6. Participant information for field trial

OFFIS Gesundheit
INSTITUT FÜR INFORMATIK

Projektleiter
Prof. Dr.
Andreas Hein

Versuchsleiter
Ralf Eckert

TELEFON
(0441) 9722-716

FAX
(0441) 9722-111

EMAIL
eckert@offis.de

OLDENBURG
11. Mai 2015

POSTANSCHRIFT
Escherweg 2
D-26121 Oldenburg

TELEFON
(0441) 9722-0

INTERNET
www.offis.de

Allgemeine Teilnehmerinformationen

Titel der Studie: Feldtest zur Evaluation eines autonom lernenden Hausautomationssystems

Herzlichen Willkommen und vielen Dank für das Interesse und die Teilnahme an unserer Studie.
Der Wunsch vieler älterer Menschen ist es, solange wie möglich im eigenen Zuhause bleiben zu können. Der Umzug in ein betreutes Wohnen oder gar in ein Heim wird solange wie möglich hinausgezögert. Doch mit steigendem Alter steigen auch die Risiken. Zum einen erhöht sich die Sturzgefahr, zum anderen besteht die Gefahr, dass gesundheitliche Veränderungen zu spät erkannt werden. Dann sind da noch die Angehörigen, die sich Sorgen um Ihre Eltern oder Großeltern machen. Zumal ist das auch die Personengruppe, die zuerst merkt, wenn sich das Alter bei Ihren Eltern/Großeltern bemerkbar macht. Leider lässt der heutige Lebensstil eine enge Betreuung durch den Angehörigen kaum noch zu. Meist leben die Familien räumlich getrennt, Arbeit und Kinder fordern schon alles von den jüngeren Angehörigen ab. Doch damit ein Mensch möglichst lange und gerne zuhause bleibt, auch wenn der Partner eventuell verstorben ist, ist Sicherheit das Wichtigste. Nur wer sich sicher fühlt kann ein selbstbestimmtes Leben führen. Das gilt für die Senioren, aber auch für die besorgten Angehörigen. Im Bereich Ambient Assisted Living (AAL), was so viel wie unaufdringlich unterstütztes Leben bedeutet, wird seit Jahren an Technik geforscht, die diese Sicherheit vermitteln soll, aber auch Notfälle erkennt oder einfach im Alltag eine Hilfe ist, ohne zu bevormunden.
Diese Studie wird im Rahmen des durch das BMBF geförderte Forschungsprojekt LivingCare durchgeführt. In diesem Forschungsprojekt soll erforscht werden, welche Unterstützung ein sogenanntes Hausautomationssystem für ältere Personen im eigenen Haushalt haben kann. Unter dem Begriff Hausautomation versteht man eine ganze Reihe elektronischer Geräte die nahezu unsichtbar in ihr Haus integriert werden. Dabei handelt es sich um Geräte, die ihre Heizung steuern können, ihre Fenster Rollläden steuern können, die Ihnen Licht einschalten, wenn Sie es brauchen, aber kein Schalter in der Nähe ist. Ein Hausautomationssystem kann vielerlei Aufgaben im Haushalt erfüllen. Bisher ging es dabei hauptsächlich um Komfort und Energieeinsparung bei Strom und Heizung. Alle Steuerbefehle, z.B. wann die Heizung in einem bestimmten Raum eingeschaltet werden soll, sind bisher an feste Regeln gebunden. Das bedeutet Sie müssen bei der Installation des Systems vorher genau festlegen, wann z.B. die Außenbeleuchtung angehen soll, wann Fenster verdunkelt werden sollen etc. Danach wird das System genauso handeln. Änderungen an diesem System müssen dann Sie oder ein Techniker vornehmen, was mit Kosten und Aufwand verbunden ist. Mit dem Projektergebnis aus LivingCare soll sich das ändern, nicht Sie sagen dem System was es tun soll, sondern das System lernt aus ihrem Verhalten. Es lernt wann Sie die Heizung einschalten, es lernt welche Lampen Sie benutzen wenn es dunkel ist und sie den Fernseher ausschalten, es lernt wann Sie

zuhause sind und wann nicht. Es lernt noch viel mehr als das und ist so in der Lage die Steuerung Ihres Hauses jederzeit Ihren sich wandelnden Bedürfnissen anzupassen, ohne das Sie das System jemals selber von Hand programmieren müssen. Das ist in jedem Fall eine schöne Neuerung, aber das ist nur eine Seite der Medaille. Durch die verbauten Bestandteile einer Hausautomation können noch ganz andere Dinge erkannt werden. Zum Beispiel plötzliche Verhaltensänderungen, die auf eine Krankheit schließen lassen. Oder wenn nach einem Sturz bekannte Bewegungsabläufe ausbleiben kann das System vorsichtshalber Hilfe rufen. Wenn Sie vergessen das Bügeleisen abzuschalten oder Herd zu lange an ist, wird das erkannt. Das System weiß wann Sie zuhause sind und wann nicht und kann so auch automatisch als Alarmanlage dienen.

Zusammengefasst erhalten Sie eine komfortable Haussteuerung, die zusätzlich die persönliche Sicherheit im Eigenheim erhöht und auch noch Energie spart.

Mit einem System, wie LivingCare es vorschlägt, können Sie selbst und ihre Angehörigen etwas entspannter leben, auch wenn keine täglichen Besuche möglich sind. Sie müssen keine Angst davor haben, zu stürzen und vielleicht tagelang nicht gefunden zu werden. Vergessene Geräte werden erkannt und abgeschaltet. Feuer können frühzeitig erkannt werden und die Feuerwehr wird verständig. Krankheits- oder Alters-bedingte Veränderungen können frühzeitig erkannt werden und so zu einem schon sehr frühen Zeitpunkt mit einer Therapie zur Linderung begonnen werden.

Alles in Allem ist LivingCare ein ernstzunehmender Schritt in Richtung „Länger selbstbestimmt Zuhause leben“.

Ablauf der Studie

Wenn Sie sich für die Teilnahme entscheiden, werden wir in einem ersten Schritt das Hausautomationssystem in ihrer Wohnung installieren. Dies geschieht in Zusammenarbeit mit einem Elektroinstallateur und dauert je nach Größe Ihrer Wohnung und Aufwand ca. einen Tag. An einem zweiten Tag wird die Anlage in Betrieb genommen. Mit Ihnen zusammen wird überprüft, ob alles wie bisher funktioniert. Da es sich noch um einen Versuch handelt, würde ab diesem Zeitpunkt das System lernen und Ihr Verhalten kennenlernen. Nach ca. 14 Tagen würden wir Sie erneut besuchen und überprüfen, zu welchen Ergebnissen das System gekommen ist. Diese Ergebnisse besprechen wir mit Ihnen auf deren Sinnhaftigkeit hin. Wenn die vorgeschlagenen Regeln mit ihren Vorlieben übereinstimmen würden, wird das System nun eingeschaltet.

Das System verfügt auch über die Möglichkeit über das Internet auf Funktion überprüft zu werden. Sobald Sie merken, dass das System fehlerhaft reagiert, bitten wie Sie uns umgehend zu informieren. Wir werden uns möglichst noch am gleichen Tag um das Problem kümmern. Sollte das nicht möglich sein besteht immer die Möglichkeit das System zu deaktivieren. Zum Hausautomationssystem wird ein Tablet, im Prinzip ein kleiner Computer, mitgeliefert. Über dieses Gerät können Sie das System überwachen und wenn erforderlich abschalten.

Die Installation wird ca. XX Monate/Wochen bei Ihnen im Haushalt verbleiben. Nach dieser Zeit werden wir das System abbauen und aus Ihrer Wohnung entfernen. Da es sich noch um einen Prototypen handelt besteht

nicht die Möglichkeit die Gerätschaften nach dem Feldtest zu übernehmen.

Datenschutz

Datenschutz ist ein sensibles Thema und genauso wird es im Projekt LivingCare auch behandelt. Daher wurde schon lange vor dem Feldtest ein umfangreiches Datenschutzkonzept erstellt.
Während des Studie zeichnen die in Ihrem Haushalt verbauten Geräte ständig Daten auf. Bewegungsmelder erfassen ihre Bewegung, die Lichtschalter merken sich wann sie gedrückt wurden, der Fernseher meldet seine Benutzung die Technik erkennt wann Sie Fenster oder Türen öffnen und schließen und wann Sie es gerne wie warm haben wollen über ihre individuelle Heizungsbenutzung. Einzelne Daten haben keinen großen Informationsgehalt. Da aber über einen längeren Zeitraum Daten aufgezeichnet werden entstehen detaillierte Profile ihres Verhaltens in der Wohnung. Wenn ein Mensch auf die Daten schauen würde, könnte er erkennen, wann Sie ins Bett gehen, wie lange Sie Fernsehen, wann Sie kochen und vieles mehr. Dinge von denen Sie höchstwahrscheinlich gar nicht möchten, das jemand diese weiß.
Aus diesen Gründen werden wir ihre Daten schützen. Die Daten der einzelnen verbauten Geräte werden auf einer Box gesammelt. Auf diese Box kann nur durch ein geheimes Passwort zugegriffen werden. Ein paar Personen müssen im Rahmen der Studie auf diese Box zugreifen:

- Sie selbst: Sie können jederzeit alle Daten einsehen, die über Sie aufgezeichnet wurden.
- Der Studienleiter: Zur Sicherstellung eines reibungslosen Ablaufs muss dem Studienleiter Zugriff auf die Box gewährt werden.
- Externe Techniker: Zu Testzwecken müssen Techniker über das Internet auf die Box zugreifen können, dafür muss der Techniker aber die Daten selbst nicht einsehen. Die Techniker kommen auch aus dem Forschungsprojekt und sind Ihnen daher nicht fremd.
- Jeder von dem Sie möchten das er/sie die Daten einsehen kann: Falls Sie möchten, dass Ihre Tochter oder Ihr Sohn die Daten einsehen können, können Sie das entscheiden und die Daten einsehbar machen.

Die Box wird auch an das Internet angeschlossen. Dieser Zugang wird besonders geschützt. Von Außen dürfen nur autorisierte Personen auf die Box zugreifen. Die Verbindung ist mehrfach gesichert:
- Eine Verbindung kann nur durch ein entsprechendes Passwort hergestellt werden.
- Die Verbindung wird nach 10 Minuten automatisch getrennt.
- Die Verbindung erfolgt verschlüsselt. Das bedeutet, die Daten sind nicht lesbar für Dritte, die nicht über einen entsprechenden Schlüssel verfügen.

Für den Betrieb des Systems sind auch nicht alle Daten erforderlich die in dem Rahmen über Sie gesammelt werden. Automatisierte Routinen werden überflüssige Daten regelmäßig löschen.

Im Rahmen von Forschungsarbeiten werden Ihre Daten weiterverendet werden. Aber selbstverständlich werden die Daten dafür anonymisiert, d.h.

es ist nicht mehr möglich aus den Daten auf ihre Person zu schließen. Die Daten werden nur zu Forschungszwecken und für die Lehre von Studenten verwendet.

Wenn Sie der Teilnahme an dieser Studie zustimmen wird Ihnen vorab eine Datenschutzerklärung vorgelegt, die wir Sie bitten aufmerksam zu lesen und nur zu unterschreiben, wenn Sie alles verstanden haben, was dort aufgeschrieben wurde.

Bei Fragen können Sie sich jederzeit an den Versuchsleiter Ralf Eckert wenden.

A.7. Subsets for evaluation of sensor placement in Living Lab Experiment #1

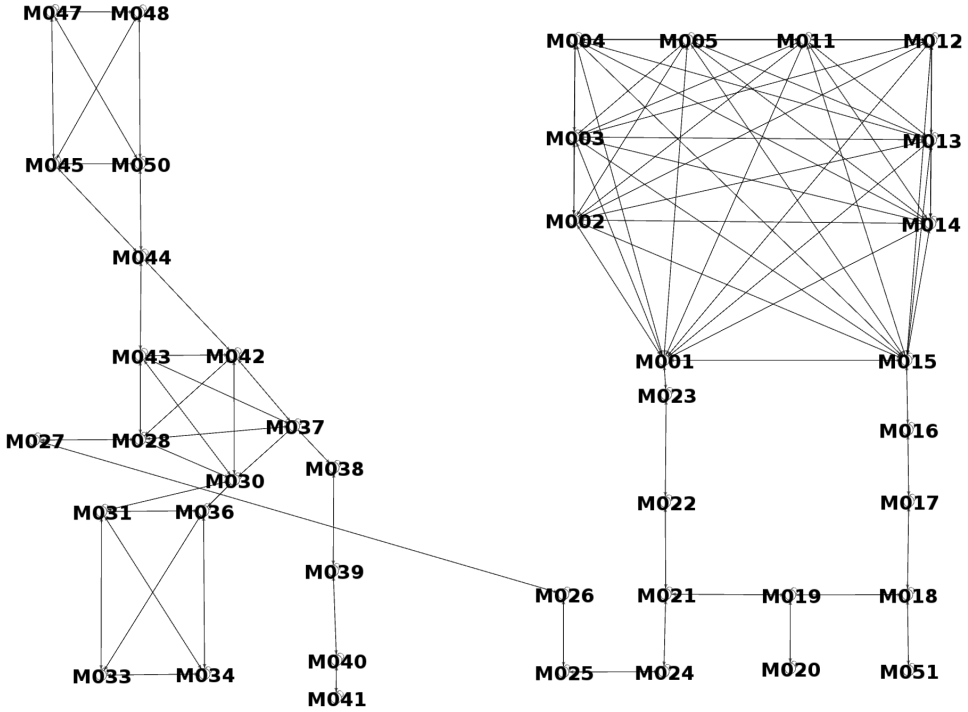


Figure A.1.: Filtering 20% of the most frequented sensors

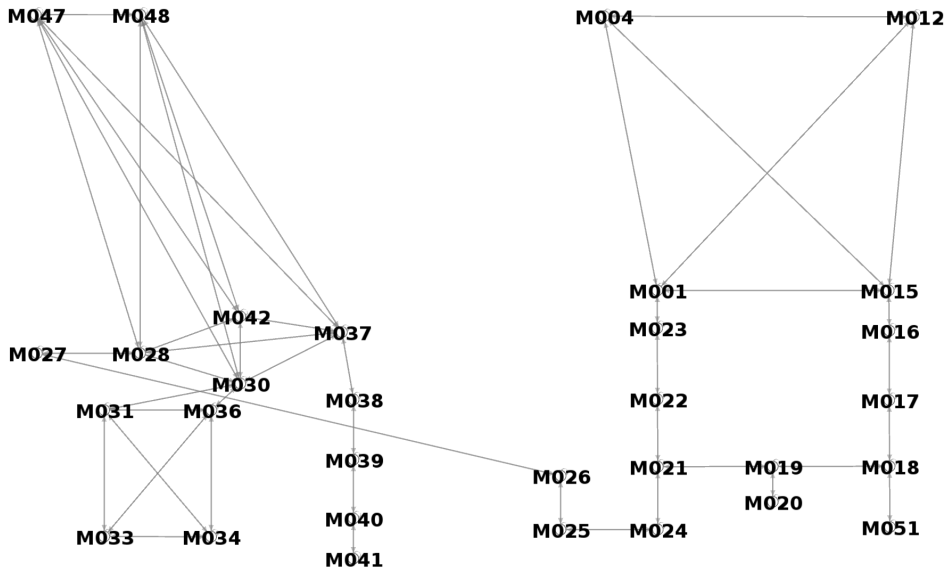


Figure A.2.: Filtering 40% of the most frequented sensors

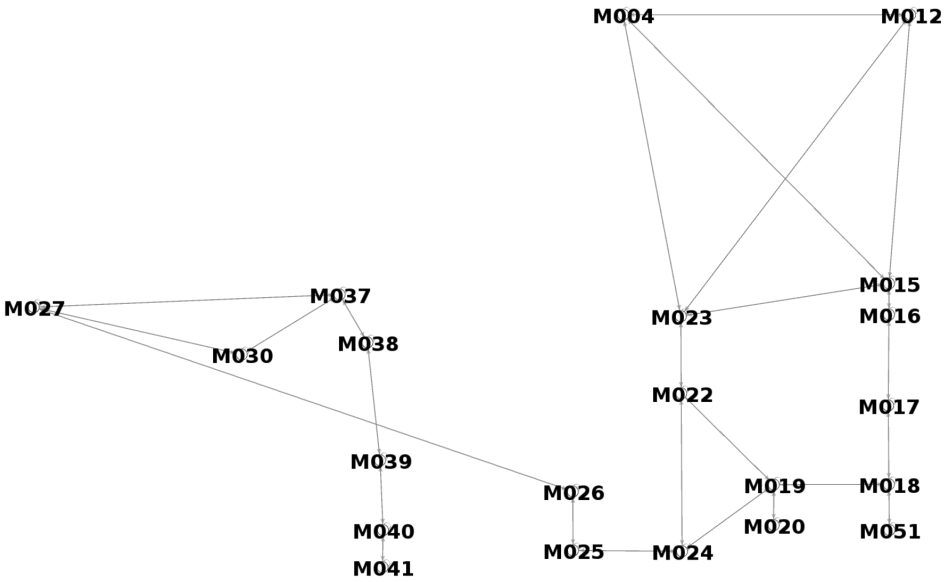


Figure A.3.: Filtering 60% of the most frequented sensors

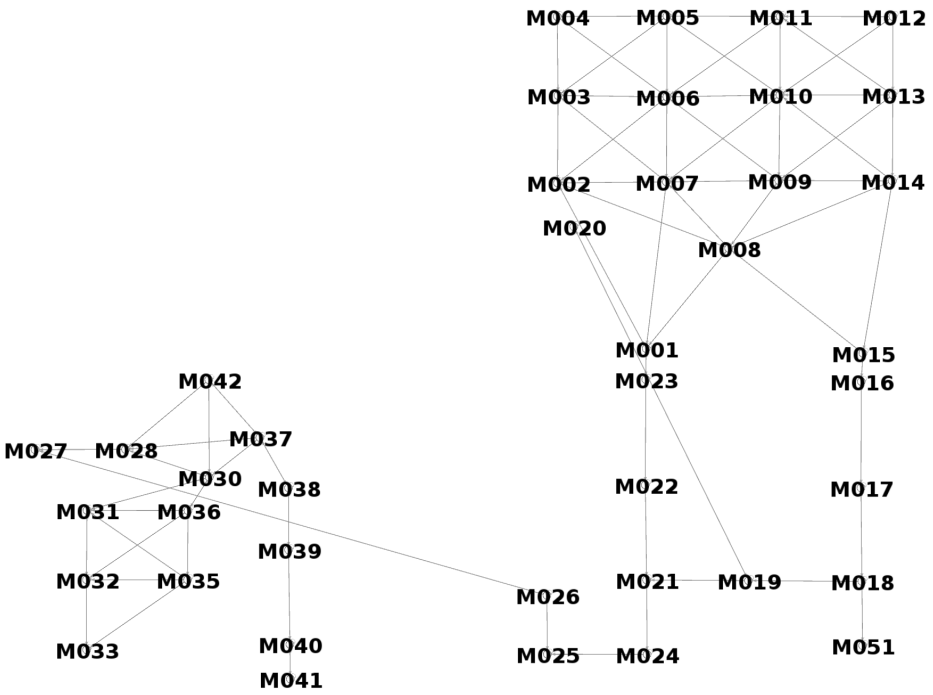


Figure A.4.: Filtering 20% of the sensors with shortest average stay

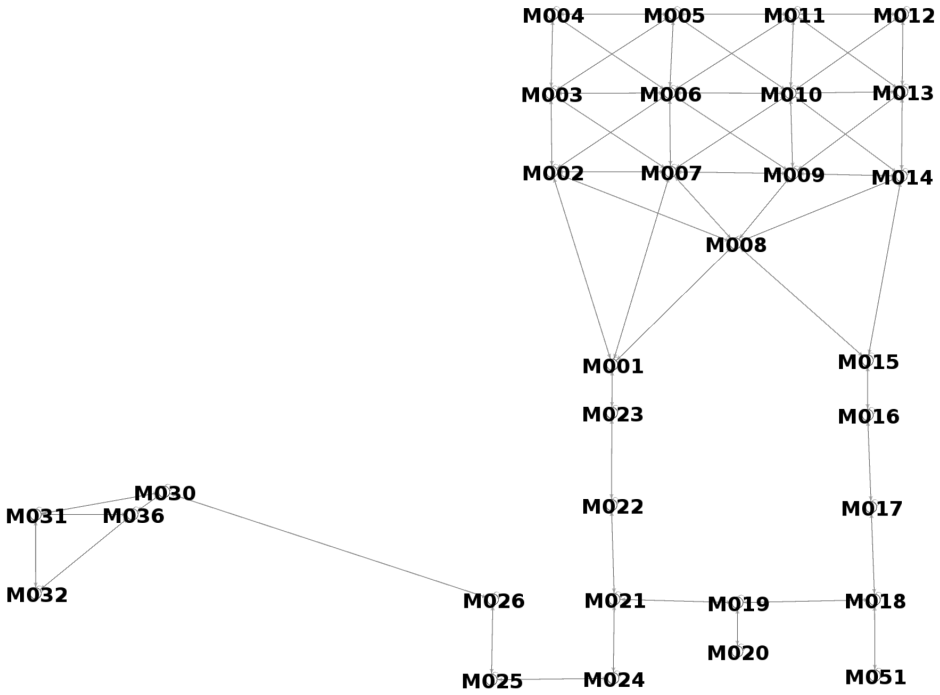


Figure A.5.: Filtering 40% of the sensors with shortest average stay

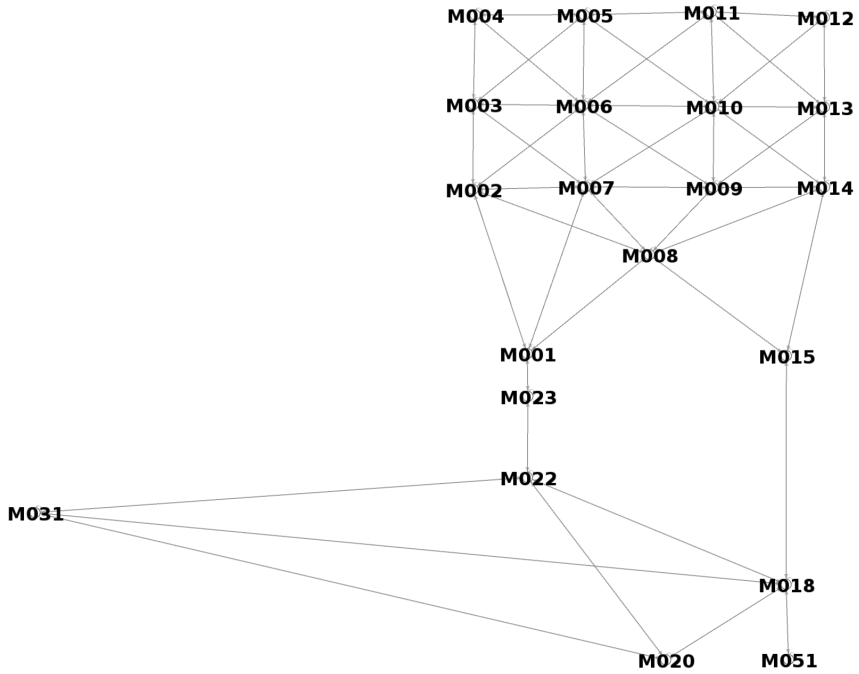


Figure A.6.: Filtering 60% of the sensors with shortest average stay

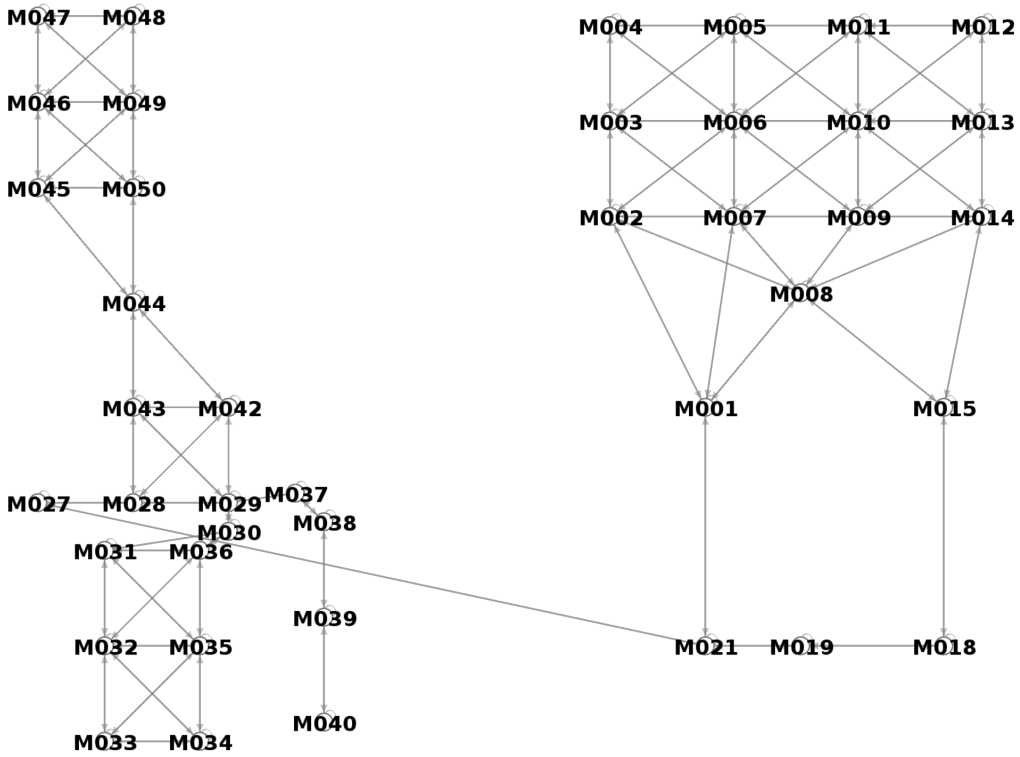


Figure A.7.: Filtering 20% of the sensors with fewest neighboring sensors

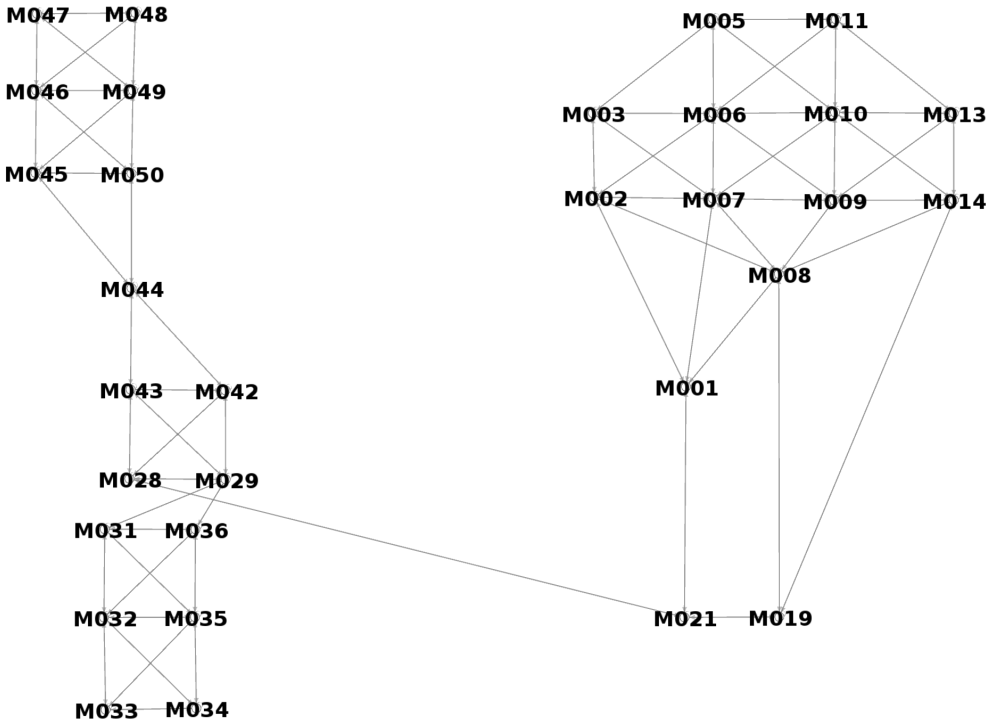


Figure A.8.: Filtering 40% of the sensors with fewest neighboring sensors

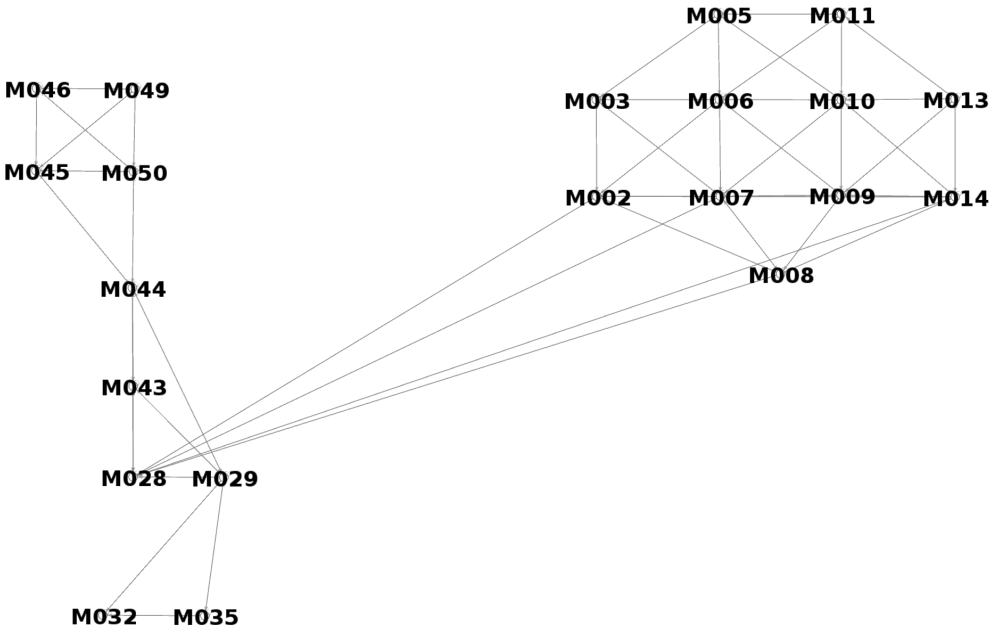


Figure A.9.: Filtering 60% of the sensors with fewest neighboring sensors

Sebastian Matthias Müller

Curriculum Vitae

Research Interests

Web3, Data Mining, Machine Learning, Smart Contracts, Multimodal Human-Computer Interaction.

Employment History

February 2019 – December 2019
"Promotionsstipendium" (Doctoral Stipend) at the OFFIS Institute for Information Technology (Oldenburg, Germany).

since February 2019
Self-employed researcher and developer on Web3, Ethereum and Smart Contracts

since May 2018
Lead Back-end and Smart Contract Developer at Flex Dapps, Melbourne, Australia.

November 2015 – February 2018
Researcher at the Institute of Distributed Autonomous Systems and Technologies at the Leuphana University Lüneburg (Lüneburg, Germany), project "LivingCare".

September 2012 – February 2018
Researcher at the OFFIS Institute for Information Technology (Oldenburg, Germany).

April 2011 – August 2012
Researcher at Tampere Unit for Computer-Human Interaction (TAUCHI) at the University of Tampere (Tampere, Finland).

January 2010 - July 2011
Research Assistant at Tampere Unit for Computer-Human Interaction (TAUCHI) at the University of Tampere (Tampere, Finland).

Education

2020
PhD (University of Oldenburg). Supervisor: Prof. Dr.-Ing. Andreas Hein. Title of Dissertation: Multi-Target Data Association and Identification in Binary Sensor Networks

2010

M.Sc. Interactive Technology (University of Tampere, Finland). Title of Dissertation: *Accessible RDF: Linked Data as a Source of Information for Visually Impaired Users*.

2008
B.Sc. Cognitive Science (University of Osnabrück, Germany). Title of Dissertation: *Using Decision Trees to Extract Human Readable Go Knowledge*.

Prizes & Awards

December 2019 Best Paper Award at the International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI) 2019

July 2015 Best Paper Award at the International Conference on Ambient Computing, Applications, Services and Technologies (AMBIENT) 2015