Proximitäts- und Aktivitätserkennung mit mobilen Endgeräten

Marco Maier

Dissertation
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig–Maximilians–Universität München

vorgelegt von
Marco Maier

Tag der Einreichung: 01. März 2016

Proximitäts- und Aktivitätserkennung mit mobilen Endgeräten

Marco Maier

Dissertation
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig–Maximilians–Universität München

vorgelegt von
Marco Maier

1. Berichterstatter:	Prof. Dr. Claudia Linnhoff-Popien
2. Berichterstatter:	Univ.-Prof. Dipl.-Math. Dr. Peter Reichl, M.A. St.
3. Berichterstatter:	Prof. Dr.-Ing. Ralf Steinmetz
Tag der Einreichung:	01. März 2016
Tag der Disputation:	22. Juli 2016

Eidesstattliche Versicherung

(siehe Promotionsordnung vom 12.07.11, § 8, Abs. 2 Pkt. 5)

Hiermit erkläre ich an Eidesstatt, dass die Dissertation von mir selbstständig, ohne unerlaubte Beihilfe angefertigt ist.

Marco Maier

Danksagung

Diese Dissertation ist während meiner Zeit am Lehrstuhl für Mobile und Verteilte Systeme an der Ludwig-Maximilians-Universität München entstanden. In dieser Zeit habe ich in unterschiedlichster Form Unterstützung und Motivation von verschiedenen Personen erfahren, denen ich an dieser Stelle gerne danken möchte.
Mein ganz besonderer Dank gilt Frau Prof. Dr. Claudia Linnhoff-Popien, die mir an ihrem Lehrstuhl ein vielfältiges und inspirierendes Arbeitsumfeld geboten hat. Insbesondere möchte ich mich für das stets entgegengebrachte Vertrauen und die große Unterstützung bei der Verfolgung meiner Interessen und Ziele bedanken.
Weiterhin gilt mein Dank Herrn Prof. Dr. Peter Reichl für die Übernahme der Zweitberichterstattung und die kritische und motivierende Diskussion meiner Arbeit, ebenso wie Herrn Prof. Dr.-Ing. Ralf Steinmetz für die Erstellung des Drittgutachtens und Herrn Prof. Dr. Christian Böhm für sein Mitwirken als Vorsitzender der Prüfungskommission.
Ein großes Dankeschön geht an alle meine Kollegen am Lehrstuhl. Der Zusammenhalt und die gegenseitige Unterstützung, die in diesem Team gegeben waren, sind nicht selbstverständlich und werden mir für immer in Erinnerung bleiben. Hervorheben möchte ich an dieser Stelle Dr. Michael Dürr, der mich für die wissenschaftliche Arbeit begeistert hat, Chadly Marouane, der mir in zahlreichen bis spät in die Nacht reichenden Diskussionen zur Seite stand, und Florian Dorfmeister, mit dem ich seit dem ersten Tag unseres Studiums viele Höhen und Tiefen dieses Weges gemeinsam durchlebt habe.
Mein größter Dank gilt jedoch meiner Familie und meiner Freundin Sarah, die in den letzten Jahren oft auf mich verzichten mussten und mich trotzdem auf diesem Weg immer unterstützt haben.

Zusammenfassung

Mit der immer größeren Verbreitung mobiler Endgeräte wie Smartphones und Tablets aber auch am Körper getragener Technik (Wearables), ist die Vision einer ubiquitär von Computern durchzogenen Welt weitgehend Realität geworden. Auf Basis dieser überall verfügbaren Technologien lassen sich mehr und mehr kontextbezogene Anwendungen umsetzen, also solche, die ihre Diensterbringung an die aktuelle Situation des Benutzers anpassen.
Ein wesentliches Kontextelement ist dabei die Proximität (Nähe) eines Benutzers zu anderen Benutzern oder Objekten. Dabei ist diese Proximität nicht nur rein örtlich zu verstehen, sondern ihre Bedeutung kann auf sämtliche Kontextelemente ausgedehnt werden. Insbesondere ist auch die Übereinstimmung von Aktivitäten verschiedener Benutzer von Interesse, um deren Zusammengehörigkeit abzuleiten. Es existiert gerade im Hinblick auf örtliche Nähe eine Reihe von Standardtechnologien, die eine Proximitätserkennung grundsätzlich erlauben. Alle diese Verfahren weisen jedoch deutliche Schwächen im Hinblick auf Sicherheit und Privatsphäre der Nutzer auf.
Im Rahmen dieser Arbeit werden drei neue Verfahren zur Proximitätserkennung vorgestellt. Dabei spielen die Komponenten „Ort“ und „Aktivität“ jeweils in unterschiedlichem Maße ein wichtige Rolle. Das erste Verfahren benutzt WLAN-Signale aus der Umgebung, um sichere, d.h. unfälschbare, Location Tags zu generieren, mit denen ein privatsphäre-schonender Proximitätstest durchgeführt werden kann.
Während das erste Verfahren rein auf örtliche Nähe abzielt, berücksichtigt das zweite Verfahren implizit auch die Aktivität der betrachteten Benutzer. Der Ansatz basiert auf der Auswertung und dem Vergleich visueller Daten, die von am Körper getragenen Kameras aufgenommen werden können.
Die Grundidee des dritten Verfahrens besteht darin, dass auch rein auf Basis von Aktivitäten bzw. Aktivitätssequenzen eine kontextuelle Proximität zwischen verschiedenen Nutzern festgestellt werden kann. Zur Umsetzung dieser Idee ist eine sehr feingranulare Aktivitätserkennung notwendig, deren Machbarkeit in dieser Arbeit ebenfalls gezeigt wird.
Zusammengenommen werden in der vorliegenden Arbeit mehrere Wege aufgezeigt, unterschiedliche Arten von kontextueller Proximität auf sichere und privatsphäre-schützende Weise festzustellen.

Abstract

With the now widespread usage of mobile devices such as smartphones and tablets as well as body-worn technical gear (Wearables), the vision of a world in which computing resources are ubiquitously available has become reality. Based on these pervasively available technologies, context-aware applications, i.e., applications adapting their provided services to a user's current situation, are becoming more and more feasible.
A primary element of a user's context is the proximity of the user to other users or objects. Proximity should not only be considered in a spatial manner but its meaning can be broadened to comprise any context element. In particular, the similarity of different users' activities is an important information to infer their contextual closeness. With regard to spatial proximity, there is a range of standard technologies which on principle allow to perform proximity detection. However, they all face severe problems with regard to security and privacy of the participants in the proximity test.
In this work, three new approaches for proximity detection are presented. Within the newly introduced systems, the contextual components „location“ and „activity“ are considered with different importance. The first approach uses Wi-fi signals from the surroundings to construct secure, i.e., unforgeable location tags, with which a privacy-preserving proximity test can be performed.
While the first method is exclusively focused on spatial proximity, the second approach also implicitly considers the users' activities. This technique is based on analyzing and comparing visual information obtained from body-mounted cameras.
The basic idea of the third approach is that contextual proximity can also be obtained based on activities alone. By comparing sequences of activities, the proximity between participating users can be inferred. In order to be realizable, this approach needs very fine-grained activity recognition capabilities. The feasibility of the latter is also shown in this work.
Summing up, in this work several ways are shown how to detect contextual proximity in a secure and privacy-preserving manner.

Inhaltsverzeichnis

1 Einleitung

> „But the world of information
> and technology doesn't always
> evolve linearly. Radical new uses
> of portable information technology
> are on the horizon."
>
> Mark Weiser, 1993

Mit der Vision, dass sich die Welt der Information und Technologie nicht immer linear, sondern von Zeit zu Zeit auch radikal verändert, begründete Mark Weiser 1993 die Idee des *Ubiquitous Computing* [189, 190, 188], also die Idee, dass sich der Mensch nicht mehr mit einem primären Eingabegerät manuell der Fähigkeiten eines Computers bedient, sondern vielmehr in einer von Computern durchsetzten Welt frei und natürlich agiert und die Technik automatisch, ohne aktiven Eingriff des Nutzers, Leben und Arbeit der Menschen unterstützt. Bereits in obigem Zitat klingt an, dass die spannenden Weiterentwicklungen vor allem im Bereich mobiler System zu erwarten sind.
Heute, mehr als 20 Jahre später, klingen zwar viele von Weisers Aussagen nach wie vor nach Zukunftsmusik, die technologische Welt hat sich jedoch tatsächlich in die skizzierte Richtung entwickelt. Bereits 2013 besaßen einer UN-Studie zufolge sechs Milliarden Menschen ein Mobiltelefon [156], bis zum Jahr 2020 soll die selbe Anzahl und damit mehr als 80% der Weltbevölkerung sogar ein *Smartphone* besitzen [122]. Die Umgebung eines Menschen ist also mittlerweile tatsächlich nahezu *ubiquitär* mit Computern und Technologie ausgestattet, und zwar einfach dadurch, dass die meisten Menschen – sowohl man selbst als auch die Menschen um einen herum – ein oder mehrere mobile, mehr oder weniger leistungsfähige Endgeräte bei sich tragen, und zwar vom Aufstehen bis zum Zubettgehen [171]. Neben Smartphones erhalten aktuell diverse Kategorien von *Wearables* – also Endgeräten, die irgendwo am Körper getragen werden – wie beispielsweise *Datenbrillen* mit integrierter Sensorik und Kamera [71], *Smart Watches* [9], *Action-Kameras* [76] oder *Fitness-Tracker* [52] eine immer größere Aufmerksamkeit und Verbreitung.
Die technischen Voraussetzungen zur Umsetzung des Ubiquitous Computing Paradigmas scheinen also gegeben, während andererseits die Idee einer automatischen Diensterbringung und quasi unsichtbaren Nutzerschnittstelle noch nicht zur Realität gehört. Gerade erst wurde der Begriff „Smombie" zum Jugendwort des Jahres 2015 in Deutschland gewählt, der als Kunstwort – zusammengesetzt aus „Smartphone" und „Zombie" – eine Person beschreibt, die

völlig gebannt auf das Smartphone blickend durch die Gegend spaziert, ohne von der Umwelt noch etwas mitzubekommen.
Trotz dieser zugegebenermaßen etwas überspitzt dargestellten Beobachtung, hat sich die Technik auch bzgl. automatischer und adaptiver Diensterbringung ernorm weiterentwickelt. Letztendlich stellt das Forschungsgebiet der *kontextbezogenen Dienste* [161, 2] die Grundlage dafür dar, Maschinen die Möglichkeit zu geben, den Kontext, d.h. die Situation, in der sich eine Person befindet, zu erfassen und darauf passend zu reagieren. Die typischste und am weitesten verbreitete Kategorie stellen dabei *ortsbezogene Dienste* (engl.: *Location Based Services (LBSs)*) dar, welche die Diensterbringung an den Standort des Benutzers anpassen, um beispielsweise die Relevanz von Suchergebnissen unter Berücksichtigung der Entfernung des gefundenen Eintrags (z.B. eines Restaurants) vom aktuellen Standort des Nutzers zu bestimmen.
Ein Grund für die Gebanntheit, mit der Menschen heutzutage in ihr Smartphone vertieft sind, und für die Menge an Zeit, die sie damit oder vor einem Laptop oder Tablet verbringen, kann auch darin liegen, dass sich die Inhalte der digitalen Welt verändert haben. Während Computer ursprünglich primär als Hilfsmittel und „Rechenmaschinen“ zur Erledigung von Arbeitsaufgaben Einsatz fanden, hat sich in der Zeit seit der Entwicklung des *World Wide Web (WWW)* [22] über das Aufkommen des *Web 2.0* [43, 148] bis hin zur aktuellen globalen Dominanz der *Online Social Networks (OSNs)*, allen voran *Facebook* [49], der Fokus der Technik immer mehr auf den Menschen selbst verschoben. Das „echte“ Leben findet also zu einem großen Teil mittlerweile (auch) digital statt, die Technik durchdringt unsere sozialen Interaktionen. Dies ist die Ausgangsbasis der vorliegenden Arbeit, deren Inhalte im Folgenden im Überblick dargestellt werden.

1.1 Motivation und Beiträge der Arbeit

Die Beobachtung, dass der Mensch als Individuum zunehmend in den Mittelpunkt der digitalen Welt rückt und gleichzeitig unser soziales Leben mehr und mehr durch Technologie begleitet wird, bringt zwei Kernherausforderungen in Bezug auf kontextbezogene Dienste – und damit in der großen Vision für das Paradigma des Ubiquitous Computing – mit sich:

i) Wir müssen technologisch in der Lage sein, soziale Interaktionen und Strukturen möglichst automatisiert zu erfassen, und

ii) wir müssen uns der Macht und des Missbrauchspotentials bewusst sein, die diese technologischen Möglichkeiten mit sich bringen, und geeignete Maßnahmen und Techniken schaffen, die die *Sicherheit* und *Privatsphäre* der Nutzer gewährleisten.

Eine Kerninformation bei der Betrachtung sozialer Beziehungen besteht darin, mit wem wir uns in der Realität treffen, mit wem wir also z.B. gemeinsam

Sport machen, Essen gehen oder durch die Stadt bummeln. Es ist intuitiv einsichtig: Die Personen, mit denen wir uns sehr oft in einer gemeinsamen Situation befinden, also eine *Gruppe* bilden, sind auf irgendeine Art und Weise wichtig für uns selbst. Andersherum bewerten wir örtliche Nähe – auch aus rein praktischen Gründen – in Bezug auf Fremde als wichtige Eigenschaft, um die Relevanz des jeweils anderen für unsere aktuelle Situation zu gewichten.

Applikationen und Dienste wie „SocialRadar“ [12], „Highlight“ [11] oder „Anomo“ [10] ermöglichen es, informiert zu werden, wenn sich *Freunde* in der Nähe des eigenen Standorts befinden, oder neue Kontakte mit Personen in der Nähe zu knüpfen. Selbst Facebook bietet mit der „Friends Nearby“-Funktion [50] eine entsprechende Möglichkeit. Eine enorme Verbreitung erleben derzeit ortsbezogene *Dating-Applikationen*, die einem Nutzer eine Auswahl potentieller Partner für ein Treffen anhand der aktuellen Entfernung [180] oder eines gemeinsamen Aufenthalts am selben Ort in der Vergangenheit [85] vorschlagen. Doch nicht nur die Nähe zu Personen, sondern auch die Nähe zu bestimmten Orten oder Objekten kann ein Kernelement eines Dienstes sein. So ermöglicht die Plattform „Foursquare“ beispielsweise das „Einchecken“ an bestimmten Orten (z.B. Restaurants), wenn man sich dort (d.h. in einer bestimmten maximalen Entfernung) befindet [54].

Während in all diesen Beispielen die örtliche *Proximität* zwischen Nutzern oder allgemein Entitäten benutzt wird, um z.B. eine Vorauswahl zu treffen oder bestimmte Aktionen vorzuschlagen oder freizuschalten, wird in der Wissenschaft im Rahmen der Forschungsgebiete der *Contextual Social Networks* [104, 96, 25] bzw. *Context-centric Social Networks* [193] und des *Pervasive Social Computing* [164] die Verbindung aus sozialen Beziehungen und kontextueller Nähe – primär des Ortes, aber auch der Aktivitäten und weiterer Kontextelemente – betrachtet und miteinander in Beziehung gesetzt. So können beispielsweise Verbindungen in einem *sozialen Graphen* eines OSN auch automatisch erzeugt werden, indem die Situationen, in denen sich die Benutzer in einem gemeinsamen Kontext befinden, als Indiz für eine Verbindung zwischen ihnen betrachtet werden.

Die Erkennung örtlicher und auch allgemein kontextueller Proximität stellt also in vierlei Hinsicht eine wichtige Fähigkeit zur Umsetzung von Anwendungen in der Schnittmenge kontextbezogener und sozialer Dienste dar. Aktuell verwendete Verfahren basieren in der Regel rein auf dem Abgleich geographischer Nutzerpositionen, die durch ein *Positionierungssystem*, z.B. GPS, ermittelt werden. Dies hat mehrere Nachteile. Grundsätzlich wird dabei nur die örtliche Proximität mit einer gewissen Genauigkeit erfasst, andere Kontextelemente wie z.B. die Aktivität der Benutzer fließen nicht mit ein.

Schwerer wiegen aber die Probleme hinsichtlich Sicherheit und Privatsphäre dieser einfachen Verfahren. Es ist zum einen nicht sichergestellt, dass die in der Regel von den Benutzern selbst übermittelte Position tatsächlich die aktuelle Position des Benutzers ist. Das Einschleusen gefälschter Positionsangaben ist in der Regel einfach zu realisieren [74, 75] und erlaubt das Vortäuschen

einer nicht vorhandenen Proximität. Die Konsequenzen können von einfachen Qualitätseinbußen bei der Benutzung eines Dienstes bis hin zu ernsthafter Beeinträchtigung von Sicherheit und Privatsphäre der Nutzer reichen, falls bestimmte Funktionen oder Informationen aufgrund der falschen Proximitätsannahme freigegeben werden.

Zum anderen setzen die bisher verwendeten einfachen Verfahren voraus, dass die Benutzer ihre aktuelle Position in Form absoluter Koordinaten zur Feststellung der Proximität freigeben. Das bedeutet, dass entweder ein zentraler Dienstanbieter und sogar alle anderen Nutzer, mit denen die Proximität geprüft werden soll, eine Historie der Aufenthaltsorte eines Nutzers anlegen und zu beliebigen Zwecken weiterverwenden können, egal ob die beiden Benutzer sich in Proximität befanden oder nicht. Selbst die „Friends Nearby"-Funktion von Facebook teilt den Freunden eines Nutzers nur mit, ob eine örtliche Nähe besteht, und nicht wo genau ein Benutzer sich auffhält, da dies von Benutzern nicht gewollt ist [39]. Da jedoch Facebook als zentraler Dienstanbieter selbst natürlich sämtliche Positionsangaben sammeln kann, ist diese Variante keine wirkliche Lösung.

Auf dieser Basis aufbauend und im Hinblick auf die erwähnten Problemstellungen, werden in der vorliegenden Arbeit im Wesentlichen drei Ansätze bzw. Verfahren vorgestellt, die auf unterschiedliche Art und Weise eine Erkennung von örtlicher bzw. kontextueller Nähe ermöglichen und dabei die Herausforderungen bzgl. Sicherheit und Privatsphäre der Nutzer in den Mittelpunkt stellen. Die Verfahren unterscheiden sich dabei nicht nur in den betrachteten Voraussetzungen und verwendeten Basis-Technologien, sondern insbesondere auch hinsichtlich ihres Fokus auf verschiedene Bestandteile kontexttueller Proximität.

Das erste vorgestellte *Proximitätserkennungsverfahren* trägt den Namen „ProbeTags" und basiert auf der Erstellung orts- und zeitspezifischer *Location Tags*, d.h. auf „Fingerabdrücken" des Ortes, mit Hilfe von WLAN-Management-Frames. Diese Fingerabdrücke enthalten zum einen keine dedizierte Information über den Ort an sich, wahren also weitgehend die Privatsphäre der Nutzer, und sind zudem nicht fälschbar, können also weder aus alten Daten wiederholt noch für die Zukunft berechnet oder vorhergesagt werden. Das Verfahren erlaubt daher eine sichere und privatsphäreschonende Erkennung von örtlicher Proximität, vor allem im sehr nahen Bereich, sodass räumliche Konstellationen, die auf Gruppensituationen hindeuten, erkannt werden können. Zudem erlaubt das Verfahren auch eine Abschätzung größerer Distanzen im Bereich mehrerer Hundert Meter. Das Verfahren wird umfangreich evaluiert, um eine Einordnung der tatsächlichen Eigenschaften vornehmen zu können.

Im nächsten Schritt wird ein Verfahren vorgestellt, das nicht nur wie der ProbeTag-Ansatz eine räumliche Proximität erkennt, sondern bei dem implizit auch die Aktivitäten der Nutzer abgeglichen werden und diese Information mit in die Berechnung der – in dem Fall eher als kontextuelle Proximität zu

bezeichnenden – Nähe der Nutzer untereinander mit einfließt. Das Verfahren basiert auf der Idee, dass Menschen, die sich in einer Gruppensituation befinden, in räumlicher Nähe zueinander stehen, ähnliche Dinge tun und dadurch *ähnliche Dinge sehen*. Letzteres wird ausgenutzt, indem das Blickfeld der Benutzer, das z.B. durch Wearables wie einer Datenbrille im Stil der Google Glass [71] oder Action Kameras wie der GoPro [76] erfasst werden kann, analysiert und mit den Blickfeldern der anderen Benutzer verglichen wird. Dabei werden jedoch keine rohen, privatsphäre-kritischen Bilddaten zwischen den Benutzern ausgetauscht, sondern aus den Bildern werden Merkmalspunkte mit Hilfe des *SURF-Verfahrens* extrahiert und auf Basis dieser die Vergleiche durchgeführt. Diese so konstruierten Merkmalsvektoren, die ähnlich wie beim ProbeTag-Verfahren als – in diesem Fall visueller – Fingerabdruck der Umgebung aufgefasst werden können, enthalten keine dedizierten Informationen über den Ort und sind durch den Einfluss beweglicher Objekte wie Personen oder Fahrzeuge wiederum nur schwer zu fälschen. Auch dieser Ansatz, aufgrund des verwendeten Verfahrens zur Merkmalsextraktion „SURFtogether“ getauft, wird umfangreich in virtuellen und realen Szenarien evaluiert.
Als dritter Ansatz wird eine Idee vorgestellt, die kontextuelle Proximität von Benutzern rein auf Basis von Aktivitäten herzuleiten. Die Grundidee besteht wiederum darin, dass sich Personen, die zusammen unterwegs sind und sich also in einer Gruppensituation befinden, im wesentlichen die gleichen Aktivitäten ausüben. Betrachtet man längere Sequenzen solcher Aktivitäten, wird die Sequenz mit zunehmender Länge immer eindeutiger identifzierbar innerhalb einer größeren Gesamtpopulation. Für eine Alltagssituation wie der gemeinsamen Fortbewegung im öffentlichen Personennahverkehr (z.B. in der U-Bahn) stellt sich die Frage, wie schnell eine entsprechende Sequenz in einer Menge von Tausenden von Menschen aussagekräftig wird. Hierzu wird in einer Simulation gezeigt, inwiefern mit den aktuell bereits robust zu erkennenden Aktivitäten „gehen“ und „stehen“ eine Aussage getroffen werden kann, und vor allem, dass sich eine deutliche feingranularere Erkennung von Aktivitäten positiv auf das Potential zur Proximitätserkennung auswirkt. Im Anschluss wird weiterhin gezeigt, dass eine solche feingranulare Aktivitätserkennung – in diesem Fall von 17 verschiedenen Aktivitäten im U-Bahn-Szenario – mit aktuellen Smartphones und der enthaltenen Sensorik möglich ist.

1.2 Mögliche Betrachtungsweisen

Die in dieser Arbeit vorgestellten und neu entwickelten Algorithmen und Techniken können aus unterschiedlichen Blickrichtungen betrachtet werden. Im Vordergrund der Forschung steht oft die Entwicklung neuer Technologien, die neue Anwendungsfelder erschließen oder bisher unbekannte Funktionalitäten ermöglichen. Die offensichtlichen Beiträge liegen dabei primär in der Schaffung neuer Möglichkeiten, d.h. der Erschließung eines komplett neuen Gebietes oder der Verbesserung vorhandener Lösungen in funktionaler Hinsicht (beispielwei-

se der Verbesserung eines Positionierungssystems von fünf Metern Genauigkeit auf einen Meter Genauigkeit).

Im Gegensatz zu dieser primär funktionalen Verbesserung vorhandener Möglichkeiten kann der Fokus auch auf der Schaffung von Ersatztechnologien liegen, die vorhandene Funktionalitäten möglichst gut kopieren, und in sekundären Gesichtspunkten, also nicht-funktionalen Eigenschaften, eine Verbesserung darstellen. In vielen Fällen bezieht sich diese Verbesserung auf die Effizienz mit der ein gewünschtes Ergebnis erreicht werden kann. Neben typischen Ressourcen wie Rechen- und Speicherkapazität oder Energie stellen die „Ressourcen“ Sicherheit und Privatsphäre wichtige Komponenten dar. Viele aktuell eingesetzte Technologien im Umfeld mobiler Endgeräte und Anwendungen bieten bereits eine ausreichende Dienstgüte um die gewünschten Funktionalitäten bereit zu stellen. Dies geschieht jedoch auf Kosten der Privatsphäre der Nutzer, wodurch ein Bedarf an Alternativlösungen entsteht.

Die dritte Betrachtungsweise bezieht sich noch ein Stück mehr auf die Themen Sicherheit und Privatsphäre. So kann es sein, dass es für einen legitimen Einsatz einer bestimmten Funktion bereits ein ausreichend gutes und effizientes Verfahren gibt und eine eventuelle Alternative im Normalfall keine Verwendung findet. Ein Benutzer kann sich willentlich gegen die Nutzung einer bestimmten Funktion entscheiden und einer Anwendung die dafür nötigen Berechtigungen, z.B. zum Zugriff auf bestimmte Sensoren, nicht erteilen. Existiert nun aber ein Verfahren, die Information auch aus anderen (Sensor-)Daten herzuleiten, die zu einem ganz anderen Zweck noch zur Verfügung stehen, wiegt sich der Benutzer in falscher Sicherheit. Ein Verfahren kann also auch bei eigentlich grundsätzlicher Unterlegenheit gegenüber einem anderen Ansatz, in einer (künstlich) eingeschränkten Situation wiederum sehr wertvoll sein, z.B. eben für einen Angreifer. Beispielsweise wurde bereits erfolgreich lediglich mit Hilfe des Beschleunigungssensors eines Smartphones auf den PIN bzw. das Entsperr-Pattern der Nutzer geschlossen [13] und Sprecher sowie einzelne gesprochende Vokabeln nur mit Hilfe des Smartphone-Gyroskops erkannt [134]. Beides sind Sensoren für die in vielen Fällen keine Freigabe des Nutzers an eine App notwendig ist, um sie zu nutzen. Diese Möglichkeiten zu kennen bzw. zu erforschen ist also unerlässlich, um Gefahren für die Sicherheit und Privatsphäre von Nutzern zu erkennen.

Die drei Betrachtungsweisen sind in der Realität nicht in dieser Konsequenz abgegrenzt, sondern gehen meist Hand in Hand. Die im Rahmen dieser Arbeit vorgestellten Ansätze sind daher auch in allen drei Gesichtspunkten von Belang, wobei sich der primäre Beitrag oft auf die zweite und dritte Kategorie bezieht. Für die meisten betrachteten Anwendungsfälle existieren bereits Technologien, die jedoch im Hinblick auf die Themen „Sicherheit“ und „Privatsphäre“ deutliche Schwächen aufweisen. Hier schaffen die im weiteren Verlauf vorgestellten Verfahren eine Alternative.

1.3 Vorveröffentlichungen

Die Inhalte dieser Arbeit wurden in Teilen bereits auf internationalen Konferenzen oder in Journal-Beiträgen publiziert. Im Folgenden wird ein Überblick über diese bereits vorveröffentlichten Inhalte sowie den jeweiligen Beitrag des Autors der vorliegenden Arbeit gegeben. In den entsprechenden Kapiteln an späterer Stelle folgt nochmals eine genauere Auflistung der bereits veröffentlichten Inhaltsteile.
Für alle Veröffentlichungen gilt, dass Prof. Dr. Claudia Linnhoff-Popien – als Lehrstuhlinhaberin und Doktormutter des Autors der vorliegenden Arbeit – beratend und als Kritikgeberin bzw. Diskussionspartnerin an den jeweiligen Papern, insbesondere in denen sie als Autorin aufgeführt ist, mitgewirkt hat. Wie sich die Anteile am Inhalt der Paper ansonsten auf die jeweils beteiligten Autoren und den Autor der vorliegenden Arbeit verteilen, wird im Folgenden im Detail erläutert.

Vis-a-Vis Verification: Social Network Identity Management Through Real World Interactions [128] Diese Arbeit stellt ein Konzept zur Verifikation der Identität der Nutzer eines Systems durch persönliche Interaktion („vis-a-vis“) vor. Das präsentierte Konzept stellt einen formalen Überbau für ein tatsächlich real umgesetztes System dar. Das System entstand im Rahmen eines Projektes, in dem eine sichere Kommunikationslösung für Bildungseinrichtungen entwickelt werden sollte. Stephan A. W. Verclas wirkte dabei als Vertreter des Projekt-Kooperationspartners bei der Analyse der Anforderungen und der Überprüfung der Praxistauglichkeit des Ansatzes mit. Benno Rott war als studentische Hilfskraft an der Implementierung des Verfahrens beteiligt. Chadly Marouane lieferte Beiträge zur Einordnung und Abgrenzung des Ansatzes in das grundsätzliche Themengebiet sowie gegenüber Related Work. Zudem hatte er maßgeblichen Anteil an der Implementierung des Systems. Die Hauptinhalte, insbesondere das theoretische Konzept des Ansatzes (Kapitel IV des Papers), stammen vom Autor der vorliegenden Arbeit. Die Inhalte der Veröffentlichung sind insbesondere in Kapitel 2.3.2 enthalten, in dem das Vis-a-Vis-System als Beispiel für einen möglichen Einsatzzweck der in dieser Arbeit vorgestellten Techniken aufgeführt wird.

Vis-a-Vis: Offline-Capable Management of Virtual Trust Structures Based on Real-Life Interactions [127] In diesem Paper wurde das Konzept aus [128] um eine Offline-Fähigkeit erweitert. Chadly Marouane hat neben den Beiträgen zu Grundlagen und Related Work des Papers auch bei der kritischen Überprüfung der vorgeschlagenen Abläufe und Protokolle mitgeholfen. Die Hauptinhalte, insbesondere die erweiterten Ausführungen zum theoretischen Konzept sowie das Protokoll zur Umsetzung der Offline-Funktionalität (Kapitel V und VI des Papers) stammen vom Autor der vorliegenden Arbeit. Analog zur ersten Vis-a-Vis-Veröffentlichung finden sich Teile des Inhalts ins-

besondere in Kapitel 2.3.2 der vorliegenden Arbeit.

ProbeTags: Privacy-Preserving Proximity Detection Using Wi-Fi Management Frames [129] Diese Arbeit präsentiert den neuartigen Ansatz zur Umsetzung von Proximitätserkennung mit Hilfe von WLAN-Management-Frames. Florian Dorfmeister befasst sich in seiner eigenen Forschung ebenfalls mit Privatsphäre- und Sicherheitsfragen in kontextbezogenen Diensten und trug zur kritischen Überprüfung des Konzeptes und zur Abgrenzung gegenüber Related Work bei. Lorenz Schauer befasst sich primär mit Ansätzen zur Indoor-Positionierung und Besucherstromsteuerung (u.a. mit WLAN) und unterstützte mit seinem Wissen die technische Umsetzung des Ansatzes, insbesondere die Modifikation der Endgeräte zum Monitoring des WLAN-Verkehrs. Die Idee, das Konzept, die theoretischen Anforderungen und Eigenschaften, sowie die Evaluation (inbesondere Kapitel II, IV und V des Papers) stammen vom Autor der vorliegenden Arbeit. Die Inhalte der Veröffentlichung bilden insbesondere die Grundlage von Kapitel 4 der vorliegenden Arbeit und sind großteils dort eingeflossen.

SURFtogether: Towards Context Proximity Detection Using Visual Features [126] In dieser Veröffentlichung wird ein neuartiger Ansatz für Proximitätserkennung vorgestellt, der auf der Extraktion und dem Vergleich von visuellen Bildmerkmalen basiert. Philipp Marcus und Florian Dorfmeister trugen zur kritischen Überprüfung und Einordnung in die bestehenden Forschungsarbeiten bei. Manuel Klette half als Master-Student beim Sammeln der Testdaten und bei der Implementierung und Durchführung der Evaluation. Chadly Marouane lieferte als Experte für Bildverarbeitungs- und Computer-Vision-Algorithmen wichtige Beiträge zur technischen Umsetzung des Ansatzes. Die grundsätzliche Idee und das Konzept mit den Erweiterungen (insbesondere Kapitel 2 und 3 des Papers) stammen vom Autor der vorliegenden Arbeit. Insbesondere Kapitel 5 der vorliegenden Arbeit basiert auf dieser Veröffentlichung und enthält deren Inhalte.

Fine-Grained Activity Recognition of Pedestrians Travelling by Subway [125] Diese Arbeit stellt ein Verfahren zur feingranularen Aktivitätserkennung von Personen, die sich im öffentlichen Nahverkehr bewegen, vor. Florian Dorfmeister hat zur Einordnung gegenüber verwandten Forschungsarbeiten sowie zur Diskussion des Ansatzes beigetragen. Die Hauptinhalte, insbesondere die Idee, das Konzept sowie die Evaluation (Kapitel 2, 3, und 4 des Papers) stammen vom Autor der vorliegenden Arbeit. Die Inhalte dieser Veröffentlichung bilden insbesondere die Basis von Kapitel 6 der vorliegenden Arbeit, wobei die Inhalte hauptsächlich in das Unterkapitel 6.3 eingeflossen sind.

1.4 Überblick über die vorliegende Arbeit

Der Rest der vorliegenden Arbeit ist folgendermaßen strukturiert: In Kapitel 2 werden wichtige Grundlagen hinsichtlich der Erkennung von räumlicher und kontextueller Proximität eingeführt. Ferner werden sowohl die Rahmenbedingungen als auch die Anforderungen für die gesuchten Verfahren zur sicheren und privatsphäreschonenden Erkennung kontextueller Proximität aufgeführt. Anschließend wird in Kapitel 3 ein breiter Überblick über verwandte Arbeiten und mögliche (Teil-)Lösungen der definierten Probleme erläutert und bewertet. In Kapitel 4 wird ein Ansatz zur Erkennung räumlicher Proximität auf Basis von WLAN-Management-Frames vorgestellt und umfangreich evaluiert. Danach folgt in Kapitel 5 ein weiterer Ansatz zur Proximitätserkennung, der jedoch nicht nur räumliche sondern implizit auch kontextuelle Proximität betrachtet, und auf Basis von visuellen Merkmalspunkten arbeitet. Als drittes Proximitätserkennungsverfahren wird in Kapitel 6 ein Ansatz auf Basis feingranularer Aktivitätserkennung vorgeschlagen und zudem die Machbarkeit der feingranularen Aktivitätserkennung gezeigt.
Abschließend wird in Kapitel 7 eine Zusammenfassung der Arbeit und ein Ausblick auf zukünftige Fragestellungen und Forschungsinhalte gegeben.

2 Grundlagen der Proximitäts- und Aktivitätserkennung mit mobilen Endgeräten

In diesem Kapitel werden wichtige Grundlagen zum Verständnis der später vorgestellten Ansätze vorgestellt. Zunächst wird das Konzept der kontextbezogenen Anwendungen allgemein erläutert und es werden die für diese Arbeit wichtigen Kontextelemente der Proximität und Aktivität vorgestellt. Anschließend folgt eine genauere Abhandlung sogenannter kontextbezogener sozialer Anwendungen, die in vielerlei Hinsicht Motivation und auch Rahmenbedingung für die später präsentierten Techniken sind. Aufbauend auf diesen allgemeineren Erläuterungen folgen wichtige Grundlagen zu den zwei Kernthemen dieser Arbeit, der Proximitäts- und der Aktivitätserkennung, die beide Hand in Hand gehen wenn es um die Erkennung allgemeiner kontextueller Nähe zwischen verschiedenen Entitäten geht. Um geeignete Verfahren zur Erkennung kontextueller Proximität identifizieren zu können, werden schließlich Anforderungen und Rahmenbedingungen für solche Verfahren vorgestellt.

2.1 Vorveröffentlichungen

In diesem Kapitel wird ein „Vis-a-Vis“ genannter Ansatz als Beispiel für kontextbezogene soziale Anwendungen angeführt und rudimentär erläutert. Der Ansatz wurde bereits in zwei Papern publiziert [128, 127], zu denen der Autor der vorliegenden Arbeit die Hauptinhalte (insbesondere das theoretische Konzept und die entworfenen Abläufe und Protokolle) beigesteuert hat. Wichtige Inhalte des Konzepts werden in Abschnitt 2.3.2 beispielhaft nochmal aufgegriffen.

2.2 Kontextbezogene Anwendungen und Ubiquitous Computing

Den generellen Rahmen für die vorliegende Arbeit stellt der Forschungsbereich der *kontextbezogenen Anwendungen* dar. Die Abhandlung von Schilit et al. [161] zu diesem Thema wird in den meisten Forschungsarbeiten als älteste Referenz und Definition herangezogen. Danach handelt es sich bei kontextbezogenen Anwendungen ganz allgemein um Systeme, die den sich verändernden

Kontext eines Individuums wahrnehmen und darauf reagieren können. Interessanterweise identifizierten die Autoren damals schon die folgenden drei Fragen als primäre Aspekte des Kontexts eines Individuums:

- Wo befindet es sich?
- Mit wem befindet es sich da?
- Was ist in der Nähe?

Alle drei Aspekte weisen damit einen Zusammenhang zum Kernthema dieser Arbeit auf, der Proximitätserkennung. Insbesondere die zweite Fragestellung soll im weiteren Verlauf der Arbeit im Mittelpunkt stehen: Mit wem befindet sich eine Entität am selben Ort? Mit wem bewegt sie sich zusammen durch die Welt? Mit wem bildet sie eine Gruppe, gleicht sich also auch in der Aktivität? Natürlich erscheint auf den ersten Blick die Frage „Was ist in der Nähe?“ die noch relevantere Fragestellung für die Thematik der Proximitätserkennung, und technisch gesehen bildet diese auch das Fundament. Zum Verständnis vieler in dieser Arbeit getroffenen Entscheidungen und zum Teil auch der grundsätzlichen Motivation für die Entwicklung einiger Ansätze ist es jedoch wichtig, dass für den Autor dieser Arbeit ähnlich wie in obiger Definition von 1994, das soziale Element des Kontexts eine zentrale Rolle spielt.

Durch das rasche Wachstum des Forschungsbereiches entwickelte sich eine Vielzahl von Arbeiten und damit Auslegungen, was unter Kontext eigentlich zu verstehen ist. Um diese Menge an Ausprägungen wieder auf einen gemeinsamen Nenner zu bringen, versuchten sich Abowd und Dey an einer allgemeingültigen, nach wie vor als Referenz dienenden, Defintion für Kontext [2]. Sie verstehen dabei unter Kontext jegliche Information, mit deren Hilfe man die Situation, in der sich eine Entität befindet, näher beschreiben kann. Als Entität kommen sowohl Personen als auch Objekte in Frage.
Da den Autoren aber die sehr generische Ausrichtung und damit für konkrete Entwicklungen evtl. wenig hilfreiche Abstraktion ihrer Definition bewusst war, identifizierten sie noch eine Reihe wichtiger Kontextelemente, die sie für die Praxis als zumeist am wichtigsten erachteten. Diese umfassen die vier Elemente *Ort*, *Identität*, *Zeit* und *Aktivität*.
Damit ist also auch die zweite wichtige Komponente der vorliegenden Arbeit, die Aktivitätserkennung, als zentrale Aufgabe zur Umsetzung kontextbezogener Dienste identifiziert.

Ein weiterer grundlegender Aspekt für die in dieser Arbeit vorgestellten Ansätze ist das Konzept des „Ubiquitous Computing“ von Mark Weiser [189, 190, 188]. Das grundsätzliche Ziel besteht dabei darin, die durch moderne Technik möglichen Dienste überall in jeder Umgebung zur Verfügung und im Einsatz zu haben, dies aber möglichst, im besten Fall sogar vollständig, unsichtbar für den Benutzer. Laut Weiser ist die beste Nutzerschnittstelle die, die man gar nicht sieht.

Diese Sichtweise ist im Prinzip die Grundlage für die später noch aufgeführten Anwendungsbeispiele, für die wiederum die im weiteren Verlauf vorgestellten Konzepte zur Proximitäts- und Aktivitätserkennung gedacht sind.

Die bisher aufgeführten Definitionen sind mittlerweile gut 20 Jahre alt. Sie dienen nach wie vor als gute Grundlage, um den Forschungsbereich abzustecken und einzuordnen. Jedoch hat sich seitdem die Technik enorm weiterentwickelt, sodass viele Ideen der damaligen Zeit mittlerweile Realität geworden sind und dafür andere Gesichtspunkte eine größere Wichtigkeit erfahren.
Spätestens durch das Aufkommen der Smartphones – für den Durchschnittsnutzer wirklich sinnvoll einsetzbar seit dem ersten Apple iPhone [197] - und der flächendeckenden Verbreitung des mobilen Internets sind viele Ideen für kontextbezogenene Dienste umsetzbar geworden. Die primäre Klasse stellen dabei die ortsbezogenen Dienste, auch Location Based Services (LBSs) genannt, dar. Angebote mit Ortsbezug versuchen die Diensterbringung mit Hilfe von Ortsinformationen des Benutzers und anderer Systemteilnehmer und interessanter Punkte (Points-of-Interest (POIs)) zu verbessern bzw. zu ermöglichen. Ein typisches Beispiel sind Suchanfragen, z.B. nach einem italienischen Restaurant, die automatisch auf den aktuellen Aufenthaltsort bezogen werden und damit die für den Benutzer nächstgelegenen Lokalitäten zurückliefern.
Mit den neuen Möglichkeiten einher gehen jedoch Herausforderungen in Bezug auf die Privatsphäre der Nutzer. Die Übermittlung der eigenen Position an einen Dienstanbieter ermöglicht eine Vielzahl von Angriffen, die private und sensible Informationen von Personen offenlegen können.
Im Folgenden werden die speziell im Rahmen dieser Arbeit relevanten Kontextelemente Proximität und Aktivität sowie Identität und soziale Beziehungen näher erläutert.

2.2.1 Proximität

Eine Subgruppe der ortsbezogenen Anwendungen stellen diejenigen Anwendungen dar, deren Ortsbezug gestützt ist auf die geographische Nähe zwischen den teilnehmenden Entitäten. Es ist zur Diensterbringung also unerheblich, an welchem Ort sich eine Entität genau befindet. Wichtig ist nur, welche anderen Entitäten sich in der Nähe bzw. in welchem Abstand dazu befinden.
Prinzipiell können also sogar bereits erwähnte Beispiele wie die Suche nach dem nächstgelegenen Restaurant in diese Kategorie eingeordnet werden. Typischerweise versteht man unter *proximitätsbezogenen Diensten* jedoch eher Angebote, welche die örtliche Nähe von mehreren mobilen, meist sogar gleichartigen, Entitäten untereinander berechnen und zur Anpassung der Diensterbringung verwenden.
Das klassische Beispiel hierfür sind sogenannte *Buddy Finder*-Anwendungen [150], welche die Benutzer benachrichtigen, wenn sich – in irgendeiner Art und Weise relevante – andere Benutzer in der Nähe befinden.

2.2.2 Aktivität

Die Aktivität eines Benutzers stellt eine der primären Kontextinformationen dar, die genutzt werden kann, um einen Dienst an die aktuelle Situation des Benutzers anzupassen [2]. „Aktivität“ kann dabei je nach Anwendungsfall ganz unterschiedlich definiert sein und sich auf unterschiedliche Bereiche und Detailgrade beziehen. Aktuelle Arbeiten zielen beispielweise auf allgemeine Aktivitäten wie „Gehen“, „Stehen“ und „Treppen steigen“ [136, 175, 143], „Essen“, „Trinken“, „Sprechen“ und „Lachen“ [199] oder allgemeine Sportübungen [47] ab. In manchen Szenarien reicht die Information, ob ein Benutzer z.B. gerade Sport treibt oder nicht, für andere Anwendungen ist es wichtig, welche Sportübung genau der Benutzer gerade ausführt oder wie viele Schritte er gelaufen ist.

2.2.3 Identität und soziale Beziehungen

Ebenfalls fester Bestandteil der Kontextdefinition sind seit jeher soziale Informationen wie die Identität des Benutzers oder die Identität der Personen, mit denen er in irgendeiner Art und Weise in Beziehung steht [161, 2]. Entsprechende Anwendungen verwenden diese Informationen über die sozialen Beziehungen und Interaktionen des Benutzers, um die Diensterbringung zu unterstützen. Das typischste Beispiel für solche Dienste stellen aktuell OSNs wie z.B. Facebook [49], Google Plus [72] oder Twitter [184] dar. In diesen sozialen Netzwerken wird von den Benutzern selbst und großteils manuell ein sog. sozialer Graph gepflegt, der die Benutzer zueinander in Beziehung setzt, sodass z.B. Freundschafts- und Freunde-von-Freunden-Beziehungen genutzt werden können, um die dargestellten Informationen zu filtern und aufzubereiten. Die bereits erwähnten Buddy-Finder-Anwendungen können so z.B. unter Kombination von Orts- und sozialen Informationen den Benutzer benachrichtigen, wenn sich seine Freunde in der Nähe befinden. Informationen über Identitäten müssen auch nicht zwingend konkret sein, sondern können auch abstrakt, z.B. als Profilinformationen vorliegen. So erlauben diverse mobile und ortsbezogene Kennenlern-Applikationen wie z.B. Tinder [180] das „Matching“ mit Personen in der Nähe, angereichert durch Profilinformationen, Fotos und Informationen aus dem sozialen Graph von Facebook.
Während diese Anwendungen den Sozialbezug sehr prominent in ihrem Dienstangebot enthalten, erscheinen für die Zukunft vor allem auch Anwendungsfälle interessant, in denen die Kontextadaptivität gar nicht auf den ersten Blick ersichtlich ist. Beispielweise könnte die Empfängerliste beim Teilen eines Dokuments automatisch nach der Wichtigkeit der Kontakte vorsortiert sein, oder Transaktionen zwischen enger miteinander verbundenen Benutzern könnten automatisch als vertrauenswürdiger eingestuft werden als zwischen weniger stark verbundenen Teilnehmern.

Während Ortsinformationen mit aktuellen Endgeräten sehr einfach automa-

tisch erstellt und an einen Dienstanbieter übermittelt werden können und auch Aktivitäten durch geeignete Algorithmen mittlerweile automatisiert erkannt werden können (beide Thematiken werden später noch im Detail aufgegriffen), stellt die Beschaffung der sozialen Kontextelemente eine noch größere Herausforderung dar. Soziale Graphen wie der von Facebook werden im Wesentlichen manuell durch die Benutzer angelegt, indem sie Kontaktanfragen stellen und bestätigen. Dies widerspricht dem Grundgedanken des Ubiquitous Computing, in dem z.B. ein solcher sozialer Graph automatisiert vom System erlernt werden können sollte. Auf Basis von elektronischer Kommunikation gibt es auch dafür bereits Ansätze [42, 23]. Besonders wichtig erscheint hier aber auch die Auswertung realer, sozialer Interaktionen. Vor diesem Hintergrund wird im Folgenden das Konzept der *kontextbezogenen sozialen Anwendungen* eingeführt.

2.3 Kontextbezogene soziale Anwendungen

Im Rahmen der vorliegenden Arbeit steht eine Klasse von kontextbezogenen Anwendungen im Mittelpunkt, die sich durch ihren Fokus auf soziale Informationen und Interaktionen auszeichnet und dabei diesen Sozialbezug wiederum aus anderen Kontextelementen ableitet. Eine hohe *kontextuelle Proximität*, d.h. eine Ähnlichkeit unterschiedlicher Dimensionen der Kontexte verschiedener Entitäten, kann als Hinweis auf eine soziale Verbindung oder Interaktion zwischen den Entitäten betrachtet werden. Dabei kann es sich je nach Anwendungsfall und betrachteten Kontextelementen sowohl um eine grundsätzliche Verbindung (d.h. z.B. eine *Freundschaftsbeziehung*) als auch um eine temporäre Zusammengehörigkeit (z.B. ein Gespräch zwischen vormals Fremden) handeln.

2.3.1 Kontextuelle Proximität

Proximität wird im Allgemeinen meist rein auf die örtliche Nähe bezogen aufgefasst. Man kann die Definition auch erweitern und eine generelle *kontextuelle Proximität* betrachten. Mit dem Ort als primärem Kontextbestandteil ist räumliche Proximität ein Teil der kontextuellen Proximität. Letztere ist jedoch auch durch andere Kontextelemente beeinflusst.
Für die vorliegende Arbeit ist neben dem Ort primär die Aktivität als Kontextinformation von Interesse. Befinden sich Entitäten in räumlicher Nähe und führen zudem auch noch die gleiche Aktivität aus, so ist ihre kontextuelle Proximität höher einzustufen als die im Verhältnis zu einer dritten Person, die unbeteiligt daneben steht.
Es ist intuitiv nachvollziehbar, dass in Alltagssituationen vor allem Menschen in Gruppen eine hohe kontextuelle Proximität aufweisen und sich damit auch von räumlich sehr nahen Personen abgrenzen lassen. Mitglied der gleichen physischen Gruppe zu sein, deutet jedoch wiederum auf eine soziale Beziehung zwischen den Teilnehmern hin. Diese Information kann genutzt werden, um

automatisiert soziale Beziehungen aus dem Verhalten der Personen abzuleiten und damit einen sozialen Graph zu erstellen.
Die Kette an Schlussfolgerung lautet also in umgekehrter Reihenfolge: Personen, die zueinander in sozialen Beziehungen stehen, befinden sich auch oft in Gruppensituationen → Gruppensituationen zeichnen sich durch eine hohe kontextuelle Proximität aus → Eine hohe kontextuelle Proximität ist insbesondere durch eine hohe räumliche Proximität verbunden mit einer großen Ähnlichkeit der ausgeführten Aktivitäten induziert.
Im Großteil dieser Arbeit steht die Erkennung von „Proximität“ im Mittelpunkt, wobei hier je nach Verfahren die Aktivität eine immer größere Rolle einnimmt, d.h. Proximität meist in der Form der kontextuellen Proximität aufgefasst wird. Wie sich im weiteren Verlauf zeigen wird, sind die Grenzen zwischen räumlicher und kontextueller Proximität und des Einflusses von Aktivitäten jedoch fließend, sodass im Extremfall sogar aus Aktivitäten eine räumliche Proximität bestimmt werden kann.

2.3.2 Sozialinteraktionsgestützte Anwendungen

Betrachtet man Situationen mit großer kontextueller Proximität, so ist es sehr naheliegend, dass diese einer Gruppensituation entsprechen, d.h. die beteiligten Nutzer sich in derselben Gruppe befinden und einander wahrnehmen. Diese Information kann wichtig sein, um die Diensterbringung in *sozialinteraktionsgestützten Anwendungen* zu ermöglichen, zu unterstützen oder abzusichern.
Als konkretes Beispiel sei hier auf ein Systemkonzept namens „Vis-a-Vis“ verwiesen [128, 127]. Dieses wurde vom Autor im Rahmen eines Projektes zu Entwicklung einer mobilen Applikation für Schulen entworfen. Durch das sensible Einsatzgebiet entstehen besonders hohe Anforderungen an die Sicherheits- und Datenschutzeigenschaften der Applikation, vor allem um sicherzugehen, dass nur berechtigte Personen, deren echte Identität nachgewiesen ist, am System teilnehmen können. Das „Vis-a-Vis“-Konzept stellt dabei eine Abstraktion der in der Applikation bestehenden Vertrauensbeziehungen und deren Zustandekommen dar.
Die Details des Konzepts sind für die weiteren Ausführungen nicht von Belang, interessant ist nur die grundsätzliche Art und Weise, wie in diesem System Vertrauensbeziehungen erstellt werden. Dazu ist es notwendig, dass eine reale soziale Interaktion zwischen einer berechtigten, bereits verifizierten Person mit der neu zum System hinzuzufügenden Person stattfindet, d.h. die beiden Personen sich „vis-a-vis“ gegenüberstehen und eine bestimmte Aktion ausführen.
In der bestehenden Implementierung ist dies durch eine dedizierte, manuelle Interaktion der Personen durchzuführen. Konkret werden Informationen durch Scannen eines visuellen Codes mit der Kamera des mobilen Endgeräts vom Display des jeweils anderen Endgeräts übertragen und als Beweis verwendet.
Um nun die manuell notwendigen Schritte im Sinne des Ubiquitous Computing zu reduzieren, kann mit Hilfe eines geeigneten Verfahrens zur Feststellung

kontextueller Proximität die Gruppensituation der beiden Beteiligten automatisiert erkannt und der generelle Ablauf damit erleichtert werden.
Dadurch entstehen wiederum enorme Anforderungen an die Sicherheit, vor allem Fälschungssicherheit, solcher Proximitätserkennungsverfahren. Besonders diese Anforderungen werden daher im weiteren Verlauf der Arbeit eine zentrale Rolle einnehmen.

2.3.3 Kontextbezogene soziale Netzwerke

Während in aktuell verbreiteten OSNs wie Facebook [49] die Beziehungen zwischen den Nutzern, d.h. der soziale Graph, durch manuelles Hinzufügen von „Freunden“ entstehen, besteht die Idee der *Contextual Social Networks* [104, 96, 25] (wahlweise auch *Context-centric Social Networks* genannt [193]) darin, die Beziehungen zwischen den Nutzern anhand ihrer Kontextähnlichkeit, d.h. ihrer kontextuellen Proximität, aufzubauen. Dies kann zur Etablierung temporärer Adhoc-Beziehungen [151] ebenso verwendet werden wie zum automatisierten Aufbau des sozialen Graphen klassischer OSNs [44, 41].
In letzterem Fall könnte es auch unterstützend wirken, um die Menge der OSN-Kontakte eines Nutzers sinnvoll zu gruppieren. Das Vorhandensein sinnvoller Gruppierungen der eigenen Kontakte beeinflusst das Verhalten der Nutzer eines OSNs nachweislich (sie teilen u.a. mehr Informationen, weil sie die Empfänger genauer bestimmen können), die manuelle Erstellung dieser ist den meisten jedoch ein zu großer Aufwand [98]. Genau diese automatisierte Einordnung der Kontakte ist auf Basis von SMS- und Anruf-Historien für die Gruppierungen „Familie“, „Arbeit“ und „Freunde“ bereits gezeigt worden [81]. Noch interessanter wäre es jedoch, wenn auch tatsächliche Interaktionen im realen Leben ausgewertet werden könnten.

2.4 Proximitätserkennung

Im Folgenden werden einige Definitionen sowie grundsätzliche Rahmenbedingungen erläutert, die für den weiteren Verlauf der Arbeit im Hinblick auf die Proximitätserkennung die Grundlage bilden. Zunächst werden die Begriffe „Proximität“ und „Gruppe“ präziser spezifiziert. Anschließend werden die gängigen Methoden zur Proximitätserkennung, wie sie heute bereits benutzt werden, erläutert und deren grundsätzliche Schwachpunkte erklärt. Darauf aufbauend werden die Anforderungen an ein sicheres und privatsphäre-schonendes Proximitätserkennungsverfahren aufgestellt.

2.4.1 Proximität

Im Allgemeinen lässt sich räumliche Proximität mit Hilfe eines Grenzwertes für die maximale Distanz definieren, unterhalb dessen zwei Entitäten als „in der Nähe voneinander“ betrachtet werden. [131]

Definition 1:
Räumliche Proximität zwischen zwei Entitäten A und B ist dann gegeben, wenn für eine gewählte Distanzfunktion d und einen gewählten Grenzwert δ gilt:

$$d(\mathrm{loc}(A), \mathrm{loc}(B)) \leq \delta \tag{2.1}$$

wobei $\mathrm{loc}(A)$ die aktuelle Position von A repräsentiert. Im Rahmen dieser Arbeit sollen Positionen nur in der Ebene betrachtet werden, d.h. in der Regel ist die Position als zweidimensionales Koordinatenpaar dargestellt, folglich $\mathrm{loc} \in \mathbb{R}^2$ und damit $d : \mathbb{R}^2 \times \mathbb{R}^2 \to E, E \subseteq \mathbb{R}$ und $\delta \in \mathbb{R}$.

Eine typischerweise verwendete Distanzfunktionen ist die euklidische Distanz d_e:

$$d_e(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{2.2}$$

Je nach Anwendungsfall sind auch andere Distanzen wie die Manhattan-Distanz oder die Distanz auf einem Routing-Graphen (z.B. dem Straßennetz) sinnvoll. Je nach Granularität des Abstandsgrenzwerts bzw. dessen Größenordnung kann auch die Verwendung von geodätischen Distanzmaßen notwendig sein, z.B. auf Basis der Haversine-Formel [196]. Für die in der vorliegenden Arbeit betrachteten Abstandsgrößenordnungen ist jedoch die euklidische Distanz zur intuitiven Einordnung der räumlichen Proximität am sinnvollsten.
Die Berechnung der Proximität auf Basis von Ortsangaben als Eingabeparameter bringt den Nachteil mit sich, dass absolute Ortsangaben preisgegeben werden müssen. Bei den in dieser Arbeit vorgestellten Verfahren soll ebendieses vermieden werden. Zudem ist wie bereits angesprochen eine Verallgemeinerung der örtlichen Proximität auf eine kontextuelle Proximität, die mehr als nur den Ort beschreiben kann, sinnvoll. Die Proximitätsdefinition muss daher von den eigentlichen Orts- und Distanzangaben abstrahiert werden. Eine allgemeinere Definition von Proximität kann gegeben werden, indem für die teilnehmenden Entitäten ein generischer *Merkmalsvektor* (im Folgenden auch *Featurevektor* genannt) betrachtet und eine geeignete Metrik zur Berechnung des Abstands verwendet wird.

Definition 2:
Kontextuelle Proximität zwischen zwei Entitäten A und B ist dann gegeben, wenn für eine gewählte Distanzfunktion d_f und einen gewählten Grenzwert δ_f gilt:

$$d_f(\mathrm{feat}(A), \mathrm{feat}(B)) \leq \delta_f \tag{2.3}$$

Dabei ist feat(A) ein Featurevektor von A, d.h. eine Menge von Werten oder Komponenten, welche die Entität bzw. die Umgebung der Entität, allgemein deren Kontext, auf irgendeine Art beschreiben. Im Allgemeinen ist $d_f : \mathbb{X}^n \times \mathbb{X}^m \rightarrow F, F \subseteq \mathbb{R}$ eine Funktion, die zwei Featurevektoren auf einen reellen Distanzwert abbildet.

Es kann sinnvoll sein, eine Feature-Distanz-Funktion zu wählen, deren Wertebereich im Intervall $[0, 1]$ liegt oder sich durch Normalisierung dahingehend modifizieren lässt. Alternativ zu Distanzen können auch Funktionen zur Bestimmung von Ähnlichkeit ($\text{sim}_f : \mathbb{X}^n \times \mathbb{X}^m \rightarrow F, F \subseteq \mathbb{R}$) zweier Featurevektoren verwendet werden, d.h. solche, die hohe Werte liefern, wenn eine hohe Ähnlichkeit vorhanden ist. Bewegt sich der Wertebereich von sim_f im Intervall $[0, 1]$, so können Distanz und Ähnlichkeit leicht miteinander in Beziehung gesetzt werden durch

$$d_f(\text{feat}(A), \text{feat}(B)) = 1 - \text{sim}_f(\text{feat}(A), \text{feat}(B)) \tag{2.4}$$

Mit dieser generischeren Form der Proximitätsdefinition geht die Intuition des geographischen euklidischen Abstands (von nun an mit d_e bezeichnet) verloren. Daher kann es in vielen Anwendungsfällen sinnvoll sein, eine Beziehung zwischen dem euklidischen und dem feature-basierten Abstand herzustellen, d.h. eine Abbildung

$$m_e : F \rightarrow E \tag{2.5}$$

zu definieren, welche die Werte der Featurevektor-Distanz auf euklidische Distanzen abbildet.

2.4.2 Gruppen

Wie im vorherigen Abschnitt beschrieben, ist der Bereich innerhalb dessen zwei Entitäten als „in Proximität“ betrachtet werden, flexibel wählbar. Im Rahmen der vorliegenden Arbeit soll der Fokus auf einem sehr engen Proximitätsbereich liegen, in dem sich eine *Gruppensituation* typischerweise abspielt.
Ausgehend von einer Definition aus der Forschung über Gruppendynamiken, kann eine *Gruppe* als bestehend aus zwei oder mehr Mitgliedern, die durch eine soziale Beziehung miteinander verbunden sind, beschrieben werden [53]. Darauf aufbauend soll sich im Rahmen der vorliegenden Arbeit eine Gruppe primär durch drei Merkmale auszeichnen:

- Sie besteht aus zwei oder mehr Mitgliedern.
- Die Mitglieder befinden sich in räumlicher Proximität mit einem maximalen Abstand von 10 Metern.

- Die Mitglieder haben eine tatsächliche Beziehung zueinander, d.h. sie üben gemeinsam eine Aktivität aus.

Die in der allgemeinen Definition angesprochene „soziale Beziehung“ soll im Rahmen dieser Arbeit also durch eine hohe kontextuelle Proximität genauer spezifiziert sein. Der im weiteren Verlauf hauptsächlich betrachtete Proximitätsbereich bewegt sich damit zwischen 0 und 10 Metern.

2.4.3 Allgemeine Verfahren zur Proximitätsbestimmung

Die Bestimmung des geographischen Abstands zwischen Entitäten in einem System, und damit je nach Anwendungsfall der Test auf Proximität zwischen Entitäten, ist wie bereits beschrieben eine weit verbreitete Funktion in einer Vielzahl von Anwendungen und Diensten. Eine Proximitätserkennung ist grundsätzlich auch sehr einfach auf Basis von absoluten Ortsinformationen umzusetzen.
Die aktuell meistgenutzte Technologie stellt dabei sicher das US-amerikanische Global Positioning System (GPS) dar, das an den meisten Orten der Erde eine satellitengestützte Positionierung auf wenige Meter genau ermöglicht. Alternativ existieren weitere Global Navigation Satellite Systems (GNSSs) wie das russische Globalnaya navigatsionnaya sputnikovaya sistema (GLONASS), das europäische Galileo-System oder das chinesische BeiDou Navigation Satellite System (BeiDou-2). GNSSs sind in der Regel nur im Freien verwendbar.
In Innenbereichen gibt es zur sogenannten *Indoor Positionierung* eine Vielzahl verschiedener Technologien. Bekannte Ansätze basieren dabei auf Funknetzen wie WLAN [14, 109, 5], Bluetooth [186, 140], Infrarot [187], Ultrabreitband [172], Ultraschall [4] oder RFID [89]. In standardmäßigen mobilen Endgeräten sind dabei bisher jedoch nur WLAN-basierte Technologien und seit einiger Zeit Bluetooth-basierte Verfahren (sogenannte iBeacons [8]) zu größerer Verbreitung gelangt.
Mit Hilfe einer oder mehrerer dieser Technologien wird auf dem mobilen Endgerät die eigene Position bestimmt und dann dem anderen Teilnehmer des Proximitätstests bzw. einer zentralen Instanz, die diesen durchführt, übermittelt. Die Position kann sich dabei je nach Anwendungsfall in unterschiedlichen Koordinatensystemen befinden und muss evtl. umgerechnet werden.
Stehen die Positionen der beiden Teilnehmer zur Verfügung, kann mit Hilfe einer Distanzfunktion wie der euklidischen oder im Fall von sphärischen Koordinaten z.B. der Haversine-Formel die geographische Distanz bestimmt werden. Liegt diese, wie in Abschnitt 2.4.1 bereits erläutert, unter einer bestimmten Grenze, so hat der Proximitätstest ein positives Ergebnis.

2.4.4 Probleme der einfachen Verfahren

Die beschriebenen allgemeinen und bereits eingesetzten Verfahren haben zwei gravierende Nachteile, welche die Sicherheit und Privatsphäre der Nutzer be-

einträchtigen.
Das erste Problem besteht darin, dass die Teilnehmer des Proximitätstests ihre tatsächliche Position den anderen Teilnehmern oder einer zentralen Instanz zur Verfügung stellen müssen, um die notwendige Berechnung durchzuführen. Das heißt, dass es entweder anderen Nutzern oder der zentralen Instanz möglich ist, eine umfangreiche Historie der Aufenthaltsorte des Nutzers zu speichern und zu beliebigen Zwecken auszuwerten. Dies untergräbt die Privatsphäre der Nutzer und kann eine Vielzahl von negativen Folgen nach sich ziehen, z.B. lassen sich sensible Aufenthaltsorte gut aus Nutzertrajektorien ableiten [105].
Das zweite Problem besteht darin, dass die Teilnehmer des Proximitätstests ihre eigene Position fälschen und beliebige Koordinaten an die zentrale Instanz oder andere Teilnehmer des Proximitätstests übermitteln können. Dadurch kann zum einen eine Proximität vorgetäuscht werden und zum anderen – selbst wenn das erstgenannte Problem auf irgendeine Weise gelöst sein sollte – durch das mehrfache, gezielte Fälschen von Positionsangaben die Privatsphäre der anderen Nutzer untergraben werden. Ein bösartiger Nutzer könnte also durch gezielte Übermittlung von Positionsangaben, die zu sensiblen Orten gehören, Informationen darüber sammeln, welche Nutzer sich an diesen zu welcher Zeit aufhalten.
Durch diese beiden Schwachstellen der aktuell eingesetzten Verfahren müssen diese im Hinblick auf Sicherheits- und Privatsphäregesichtspunkte ganz klar als ungeeignet eingestuft werden. Darüberhinaus sind diese Verfahren zudem rein auf die örtliche Proximität der Benutzer ausgelegt. Die bereits angesprochene allgemeinere kontextuelle Proximität, die insbesondere auch die Aktivität der Nutzer berücksichtigt, kann damit nur unzureichend ermittelt werden.

2.4.5 Anforderungen an sichere und privatsphäreschonende Verfahren

Auf Grundlage der bereits erläuterten Anwendungsfälle für Proximitätserkennung sowie der Schwachstellen der bisher dafür eingesetzten Verfahren, werden im Folgenden die als wesentlich erachteten Anforderungen an sichere und privatsphäreschonende Verfahren zur Proximitätserkennung aufgestellt.

Unfälschbarkeit der Positionsangabe Die primäre Anforderung an das gesuchte Proximitätsverfahren stellt die Sicherstellung der Unfälschbarkeit der Postionsangabe dar. Es darf einem Benutzer nicht möglich sein, für sich selbst eine falsche Positionsangabe in den Proximitätstest einzuschleusen bzw. es darf nicht möglich sein, eine Proximität durch Einschleusen einer gefälschten Angabe vorzutäuschen. In vielen Fällen wird es schwer zu verhindern sein, dass ein bösartiger Benutzer die von ihm übermittelten Daten manipuliert. Es darf ihm jedoch nicht möglich sein, die Manipulation derart vorzunehmen, dass die Daten gezielt zu einer Position passen, an der sich der bösartige Benutzer nicht befindet.

Dies ist auch deswegen die wichtigste Anforderung, weil wie bereits beschrieben, nicht nur eine nicht-vorhandene Proximität vorgetäuscht, sondern auch durch gezieltes Einspielen bestimmter Ortsangaben die Privatsphäre anderer Nutzer beeinträchtigt werden kann.

Wahrung der Privatsphäre im Nicht-Proximitätsfall Die zweite wichtige Anforderung besteht darin, dass ein anderer Teilnehmer des Proximitätstests die tatsächliche Position des Benutzers nicht erfahren soll, wenn sich beide nicht in Proximität befinden. Auch im Fall tatsächlicher Proximität soll die absolute Position nach Möglichkeit nicht bekannt werden. Diese Information ist jedoch je nach Größe des Proximitätsbereichs in der grundsätzlichen Information über die bestehende räumliche Nähe implizit enthalten.
Führen zwei Entitäten also einen Proximitätstest durch und schlägt dieser fehl, so erfahren beide Parteien nur, dass sie sich nicht innerhalb des getesteten Proximitätsbereichs befinden, und insbesondere erfahren sie nicht, an welchem Ort sich der jeweils andere aufhält. Ist der Proximitätstest erfolgreich, so lernen die beiden Parteien im besten Fall nur, dass sie sich innerhalb des definierten Proximitätsbereichs befinden, aber nicht wo genau.

Diese beiden Anforderungen sind im Allgemeinen gültig und essentiell zur Umsetzung einer sicheren und privatsphäreschonenden Proximitätserkennung. Im Rahmen der vorliegenden Arbeit sind die grundsätzlichen Rahmenbedingungen, unter denen das Verfahren einsetzbar sein soll, zudem etwas enger gefasst. Daher ergeben sich die folgenden weiteren Anforderungen:

Erkennung von Proximitäten im Bereich von 0 bis 10 Metern Wie bereits beschrieben gilt dieser Bereich im Rahmen der vorliegenden Arbeit als der relevanteste bei der Betrachtung von Gruppensituationen bzw. sozialen Interaktionen, die wiederum wichtig sind, um die skizzierten Anwendungsfälle wie z.B. den Einsatz zur Generierung kontextbezogener sozialer Netze zu ermöglichen und zu unterstützen. Je umfangreicher und trotzdem fein abgestufter der Proximitätsbereich mit einem Verfahren bestimmt werden kann umso besser, d.h. es wird im weiteren Verlauf auch die Eignung für größere Entfernungen betrachtet. Der primäre Bereich soll jedoch bei bis zu zehn Metern liegen.

Berücksichtigung von kontextueller Proximität Das Proximitätserkennungsverfahren soll nach Möglichkeit nicht nur rein auf Basis örtlicher Proximität arbeiten, sondern auch explizit oder implizit die Aktivitäten der Benutzer mit einbeziehen. Gerade im Hinblick auf kontextbezogene soziale Netze und ähnliche Anwendungsfälle spielt die soziale Nähe unter den Nutzern, die nicht allein durch den gleichzeitigen Aufenthalt am selben Ort gefolgert werden kann, eine große Rolle.

Verzicht auf dedizierte Infrastruktur Das Proximitätsverfahren soll möglichst ohne dedizierte, d.h. extra auszubringende Infrastruktur funktionieren. Dadurch ist gewährleistet, dass das System weitflächig und ohne großen Aufwand eingesetzt werden kann.

Benutzung von Standard-Hardware Durch den Verzicht auf dedizierte Infrastruktur wird das Proximitätserkennungsverfahren in der Regel eine Komponente erfordern, die sich in irgendeiner Art und Weise bei den Endnutzern bzw. allgemein den teilnehmenden Entitäten befindet. Diese Komponente soll möglichst auf Standard-Hardware basieren und im besten Fall ein Gerät sein, das die Benutzer heute in der Regel bereits bei sich tragen bzw. in der Zukunft mit hoher Wahrscheinlichkeit bei sich tragen werden.

Um eine genauere Vorstellung eines konkreten Szenarios zu haben, sei für den weiteren Verlauf zudem das folgende generische Szenario als primärer Anwendungsfall definiert: Die Benutzer, für die die Proximität festgestellt werden soll, bilden eine Gruppe, befinden sich innerhalb des definierten Proximitätsbereichs und bewegen sich gemeinsam, primär zu Fuß, in einer öffentlichen, städtischen Umgebung fort. Sie können dabei unterschiedliche Aktivitäten ausführen und auch zeitweise an einem Ort verweilen. Die Fortbewegung schließt auch die Benutzung öffentlicher Verkehrsmittel mit ein. Eine Benutzung von PKWs oder Fahrrädern soll vorerst nicht betrachtet werden. Zudem ist der Aufenthalt in privaten oder sehr ländlichen Umgebungen ebenfalls nicht Teil des primären Szenarios.

2.4.6 Location Tags

Eine vielversprechende Komponente zur Erstellung eines Proximitätserkennungsverfahrens, das obige Eigenschaften erfüllt, könnten sogenannte *Location Tags* sein. Der Begriff der Location Tags wurde initial von Qiu et al. [153] geprägt, die jedoch eine sehr enge und bzgl. der Eigenschaften etwas andere Definition gewählt haben, als die meisten der folgenden Arbeiten. Unter anderem wurde eine Robustheit der Tags über die Zeit gefordert und es wurden nur Funksignale betrachtet.
Die gängigere Definition wurde erstmals von Narayanan et al. gegeben [144]. Darauf aufbauend soll für die vorliegende Arbeit folgende Definition gelten:

Definition 3:
Bei **Location Tags** handelt es sich um Eigenschaften der physischen Umgebung, die spezifisch für einen Ort zu einer bestimmten Zeit sind.

Die Eigenschaften können in der Praxis sämtliche physikalischen Gegebenheiten sein, die durch ein entsprechendes Modul bzw. einen Sensor gemessen werden können. Dabei sind zwei Eigenschaften der Location Tags essentiell,

damit sie für den Anwendungsfall der Proximitätserkennung geeignet sind:

- **Reproduzierbarkeit:** Für zwei Messungen, die am selben Ort zur selben Zeit durchgeführt wurden, ist es sehr wahrscheinlich, dass sie bzgl. eines geeigneten Ähnlichkeitsmaßes identisch oder zumindest sehr ähnlich sind.

- **Nicht-Vorhersagbarkeit:** Ein Messergebnis bzw. ein Location Tag an einem Ort zu einer bestimmten Zeit kann nur an diesem Ort zu genau dieser Zeit bestimmt werden. Insbesondere ist es nicht möglich, einen Location Tag durch Berechnung oder historische Daten vorherzusagen.

Narayanan et al. erklären, dass sich verschiedene physikalische Gegebenheiten zur Bildung von Location Tags eignen können, u.a. Funksignale (WLAN, Bluetooth, GPS, GSM), Audiodaten oder auch die Zusammensetzung der Luft (mit Hilfe von Sensoren, die z.B. den Stickstoffgehalt der Luft bestimmen können). Sie bleiben bei der Behandlung jedoch oberflächlich. Konkrete Ausgestaltungen von Location Tags werden später in dieser Arbeit im Rahmen der Betrachtung verwandter Arbeiten behandelt.

2.5 Aktivitätserkennung

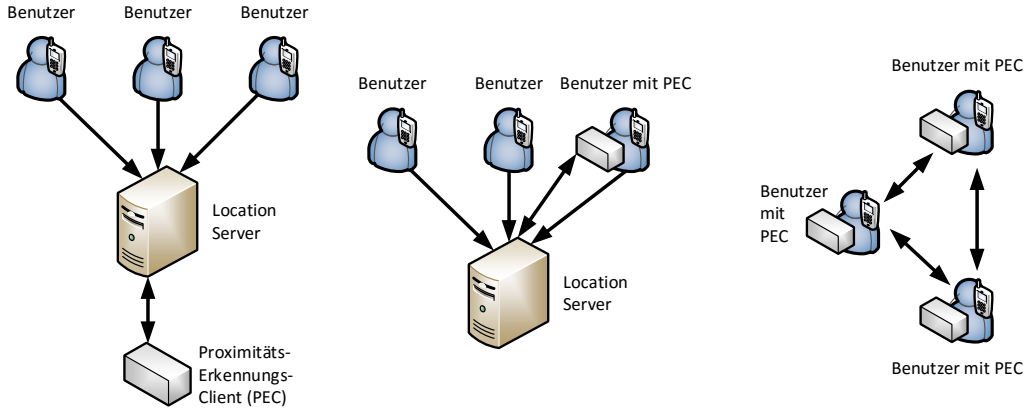
In den bisherigen Ausführungen wird das Kontextelement der „Aktivität“ oft als wichtiges Element zur Bestimmung kontextueller Proximität aufgeführt. Eine genauere Erläuterung der damit verbundenen Aktivitätserkennung soll im Folgenden gegeben werden.
Ziel dieser Verfahren ist es also, Aktivitäten von Benutzern automatisiert, meist mit Hilfe von Sensoren, zu erkennen. Dabei hängen die betrachteten Aktivitäten, deren Abstraktionsniveau und deren Granularität vom Anwendungsfall ab. Beispiele hierfür sind einfache Unterscheidungen zwischen „gehen“, „stehen“ und „sitzen“ [113], die Erkennung von alltäglichen Aktivitäten (engl: *Activities of Daily Living (ADL)*) wie „Staubsaugen“ und „Zähne putzen“ [154] oder der Bestimmung des aktuellen Fortbewegungsmittels [155, 202].
Es können im Bereich der Aktivitätserkennung zwei grundsätzliche Heransgehensweisen unterschieden werden. Die eine Kategorie der Verfahren arbeitet mit Sensoren, welche die Situation bzw. die Person(en) von außen beobachten. Oft (aber nicht ausschließlich) geschieht dies mit Videokameras, deren Bildmaterial ausgewertet wird [152]. Im Rahmen der vorliegenden Arbeit sind diese jedoch für die hier gesuchte Proximitätserkennung mit den bereits skizzierten Anforderungen weniger interessant.
Vielmehr sind Verfahren von Interesse, die Aktivitäten anhand von am Körper getragenen Endgeräten, d.h. Smartphones, Wearables oder dedizierten Sensoren, erkennen können. Das grundsätzliche Vorgehen ist dabei bei den meisten Verfahren sehr ähnlich: Es werden Sensordaten gesammelt, vorverarbeitet und

in Zeitfenstern aggregiert, bevor auf diesen Fenstern dann (statistische) Merkmale berechnet und zur Erstellung eines Klassifikationsmodells mit Hilfe eines Verfahrens aus dem Bereich des maschinellen Lernens verwendet werden [111]. Eine der Hauptaufgaben und -unterscheidungsmerkmale der unterschiedlichen Verfahren liegt in der Auswahl der verwendeten Sensordaten und Merkmale. Die „Kunst“, einen gewünschten Satz an Aktivitäten zu erkennen, besteht also meist darin, geeignete Ausgangsdaten und Merkmalsfunktionen zu identifizieren.
Neben Aktivitäten einzelner Nutzer lassen sich Aktivitäten ganzer Gruppen erkennen [77, 78] oder es kann das Wissen um eine bestehende Gruppe dazu genutzt werden, wiederum die Aktivitäten einzelner genauer zu erkennen [92]. Wie in den vorherigen Ausführungen schon mehrfach angeklungen, sind die Themen Proximität und Aktivität also stark miteinander verwoben. Gruppensituationen können genutzt werden, um Aktivitäten besser zu erkennen, und gemeinsame Aktivitäten können auf Gruppen und Proximitätssituationen hindeuten. Letzteres wird im Rahmen der vorliegenden Arbeit mehrfach benutzt.

2.6 System- und Angreifermodell für Proximitätsdienste

Ein System zur Proximitätserkennung kann als spezielle Ausprägung bzw. Instanziierung eines ortsbezogenen Dienstes betrachtet werden. Angelehnt an das generische Systemmodell für ortsbezogene Dienste von Wernke et al. [195] können drei Klassen von Teilnehmern des Systems identifiziert werden: *Benutzer*, *Location Server* und *Clients*.
Benutzer sind die den Dienst in Anspruch nehmenden Entitäten, die jeweils eine entsprechende Ausstattung besitzen, um die notwendigen Informationen über ihren aktuellen Aufenthaltsort lokal zu ermitteln. Es kann sich hierbei um echte Personen handeln, die z.B. ein modernes Smartphone mit sich führen, das mit der entsprechend benötigten Sensorik bzw. Hardware ausgestattet ist. Benutzer können jedoch auch andere Objekte sein, z.B. Maschinen, die automatisiert ihre Funktion an Proximitätsinformationen anpassen sollen. Wie genau der aktuelle Aufenthaltsort beschrieben werden muss, also mit Hilfe welcher Informationen, ist an dieser Stelle bewusst offen gehalten. Im einfachsten Fall handelt es sich um absolute Ortsangaben in Form von Koordinaten. Gegenstand der vorliegenden Arbeit ist aber genau die Entwicklung und Untersuchung alternativer Beschreibungen des Aufenthaltsorts, sodass die von den Benutzern übermittelten Ortsinformationen zunächst als generisches Datenpaket betrachtet werden können.
Diese Daten werden dann an eine im ursprünglichen Modell [195] als Location Server bezeichnete zentrale Instanz übermittelt. Diese hält die Informationen vor und stellt sie bei Bedarf und je nach Anwendungsfall anhand bestimmter Zugriffsregeln den Clients zur Verfügung.

Clients sind generisch betrachtet jegliche Dienste, die die vom Location Server zu beziehenden Ortsinformationen verwerten. Da der Fokus der vorliegenden Arbeit auf der Erkennung von Proximität zwischen Entitäten liegt, kann die Funktion des Clients bereits dahingehend präzisiert werden. Der Client wird daher als Proximitäts-Erkennungs-Client (PEC) bezeichnet. Seine Aufgabe besteht darin, die Ortsinformationen vom Location Server anzufragen und anhand dieser Daten die Proximität der Benutzer untereinander zu ermitteln.
In Abbildung 2.1 sind drei verschiedene Konfigurationen der Teilnehmer visualisiert. Analog zum von Wernke et al. skizzierten Modell können die drei



(a) Grundmodell (b) Benutzer mit PEC (c) Peer-to-Peer

Abbildung 2.1: Systemmodell für Proximitätsdienste, angeleht an ein allgemeines Modell ortsbezogener Dienste von Wernke et al. [195]. Es interagieren die drei Komponenten „Benutzer“, „Location Server“ und „Proximitäts-Erkennungs-Client“, wobei diese verteilt auf unterschiedliche physische Entitäten oder teilweise integriert vorhanden sein können.

Komponenten wie in Abbildung 2.1a gezeigt als getrennte Einheiten betrachtet werden. In der Regel ist jedoch davon auszugehen, dass die Benutzer – egal ob Personen oder Maschinen oder beides – die Informationen bzgl. Proximitäten untereinander benötigen, d.h. entweder wiederum beim PEC anfragen müssen oder, wie in Abbildung 2.1b dargestellt, selbst einen PEC besitzen, d.h. anhand der Ortsinformationen vom Location Server lokal die Proximität zu anderen Nutzern bestimmen können. Denkt man diese Konfiguration noch einen Schritt weiter, so ließe sich das Gesamtsystem auch komplett Peer-to-Peer (P2P)-basiert, also verteilt ohne zentralen Location Server umsetzen. Dies ist in Abbildung 2.1c skizziert.

Angreifermodell Anhand der so definierten Teilnehmer des Systems kann nun ein Angreifermodell beschrieben werden, d.h. welche Komponenten in welchem Umfang vertrauenswürdig sind. Die Benutzer selbst sind dabei aus zwei

Blickrichtungen zu betrachten. Aus Sicht eines Benutzers auf sich selbst wird die lokale Hard- und Softwareumgebung als vertrauenswürdig angenommen, d.h. die lokale Erstellung der Ortsinformation führt zum korrekten Ergebnis. Szenarien, in denen das Endgerät des Benutzers kompromittiert ist, spielen im Rahmen dieser Arbeit keine Rolle. Betrachtet man aus Sicht eines Benutzers die anderen Benutzer des Systems, so werden diese als *nicht vertrauenswürdig* angenommen. Man muss also davon ausgehen, dass fremde Benutzer manipulierte Ortsinformationen zur Verfügung stellen, um damit beispielsweise eine nicht gegebene Proximität vorzutäuschen.

Der Location-Server, sofern vorhanden, wird soweit als vertrauenswürdig angenommen, dass er keine aktiven Manipulationen an den an ihn übermittelten Ortsinformationen vornimmt, d.h. insbesondere auch, dass er die Zuordnungen von Ortsinformation zum zugehörigen Benutzer nicht verändert. Dies kann im Zweifel durch sekundäre Mechanismen wie die kryptographische Signatur [68] der bereitgestellten Ortsinformationen durch die Benutzer sichergestellt werden. Dadurch, dass durch die Benutzer jedoch manipulierte Ortsinformationen bereitgestellt werden können, kann nicht davon ausgegangen werden, dass vom Location Server übermittelte Ortsinformationen tatsächlich vertrauenswürdig sind.

Der PEC wird im weiteren Verlauf als vertrauenswürdig angenommen, da er im Zweifel lokal von den Benutzern selbst instanziiert wird. Eine Auslagerung der Komponente auf eine dritte Partei birgt in der Praxis natürlich Risiken, wird aber aus dem beschriebenen Grund für die weitere Betrachtung nicht berücksichtigt.

Bzgl. der Validität der Ortsinformationen besteht also die grundsätzliche Gefahr der Manipulation durch die Benutzer, primär mit dem Ziel, eine nicht vorhandene Proximität vorzutäuschen. Darüberhinaus gilt aus Privatsphäresicht für alle beteiligten Komponenten das Honest-but-Curious-Modell, d.h. die Beteiligten halten sich an die Kommunikationsprotokolle und unternehmen keine aktiven Manipulationen, um mehr Informationen zu erhalten und die Privatsphäre anderer Benutzer zu beeinträchtigen. Sie sind jedoch „neugierig“, d.h. sie können die freiwillig zur Verfügung gestellten Informationen speichern und weiterverarbeiten.

Es ist zu beachten, dass die Manipulation der zur Verfügung gestellten Ortsinformationen durch die Benutzer und damit eine eventuell vorgetäuschte, nicht vorhandene Proximität zu Entscheidungen auf Anwendungsebene, z.B. der Erteilung von Berechtigungen, führen kann, die dann in der Konsequenz die Privatsphäre der betroffenen Entität beeinträchtigen. Dieser Fall betrifft jedoch Daten, die im hier betrachteten Systemmodell nicht vorhanden sind, bzw. bezieht sich auf eine Ebene des Gesamtsystems, die in obigem Systemmodell nicht enthalten ist. Es ist daher valide, für die im betrachteten Systemmodell vorhandenen Daten und Interaktionen das Honest-but-Curious-Modell bzgl. der Privatsphäre der Benutzer anzunehmen. Der Fall von manipulierten Ortsinformationen ist in vielerlei Hinsicht relevant und wird daher im weiteren Verlauf

als Sicherheits- und nicht rein als Privatsphärethematik eingestuft.

2.7 Zusammenfassung

In diesem Kapitel wurden neben einer grundsätzlichen Einführung in das Thema der kontextbezogenen Dienste und diverser Unterpunkte davon die beiden Kerninhalte „Proximitätserkennung“ und „Aktivitätserkennung“ vorgestellt. Die Idee der „kontextuellen Proximität“ stellt dabei für die nachfolgenden Kapitel ein Kernelement dar. Kontextuelle Proximität ist für den Großteil der im Alltag relevanten Situation natürlich primär durch die örtliche Nähe der betrachteten Personen geprägt, weswegen diese auch im Zentrum dieser Arbeit steht. Sie ist jedoch auch abhängig von der Ähnlichkeit der Aktivitäten der Nutzer, womit eine explizite Aktivitätserkennung oder eine implizite Berücksichtigung sinnvoll erscheint.
Um geeignete Verfahren zur Erkennung von (kontextueller) Proximität zu identifizieren, wurden Anforderungen, Rahmenbedingungen sowie ein grundsätzliches Systemmodell aufgestellt. Anhand dieser Vorgaben wird im nächsten Kapitel ein breiter Überblick über verwandte Arbeiten gegeben, bevor ab Kapitel 4 neuentwickelte Verfahren vorgestellt werden.

3 Verwandte Arbeiten

In diesem Kapitel erfolgt ein genereller Überblick über die ganze Bandbreite relevanter Ansätze im Bereich der Proximitätserkennung, sowohl über solche mit dem Schwerpunkt auf dem Schutz der Privatsphäre als auch über solche, die primär die Echtheit der Proximitätsangabe gewährleisten wollen. Verwandte Arbeiten, die nur eine spezielle Relevanz für die einzelnen in den Kapiteln 4, 5 und 6 neu vorgestellten Ansätzen besitzen, werden im jeweiligen Kapitel in einem gesonderten Überblick betrachtet. Insbesondere gilt dies für Arbeiten zur Aktitivitätserkennung, die speziell für Kapitel 6 von Bedeutung sind.

3.1 Frühe Optimierungsziele im Bereich Proximitätserkennung

Mit dem Ziel der Vollständigkeit des Überblicks sei vorab darauf hingewiesen, dass sich bzgl. der Thematik der Proximitätserkennung unterschiedliche Schwerpunkte in der Forschung entwickelt haben. Gerade in den ersten Jahren der größeren Verbreitung ortsbezogener und damit auch proximitätsbasierter Dienste entstanden viele Arbeiten, die sich primär Fragen der Effizienz der Proximitätsbestimmung widmeten. Insbesondere wurden Verfahren zur Verringerung des Kommunikationsaufwands in einem Gesamtsystem entwickelt [182, 107] und das sowohl für zentralisierte als auch für P2P-basierte Systeme [6]. Im Rahmen der vorliegenden Arbeit spielen diese Gesichtspunkte jedoch keine Rolle. Der Fokus liegt auf alternativen Methoden zur Erkennung von Proximität an sich und deren Eigenschaften vor allem bzgl. Sicherheit und Privatsphäre.

3.2 Schutz der Privatsphäre

Im Folgenden werden verschiedene Klassen von Ansätzen bzgl. des Schutzes der Privatsphäre vorgestellt. Die meisten dieser Ansätze wurden allgemein für ortsbezogene Dienste entwickelt. Wie in Kapitel 2.4.5 erläutert, bezieht sich die Privatsphäre-Thematik bei Proximitätserkennungsverfahren aber genau auf die bereitgestellten Ortsinformationen, sodass die existierenden generischen Ansätze Anwendung finden können. Die Klassifizierung orientiert sich im Wesentlichen an etablierten Einteilungen [195].

3.2.1 Privacy Policies

Mit dem Aufkommen der ortsbezogenen Dienste und damit der zunehmenden Sensibilisierung bzgl. der Privatsphäre-Thematik in diesem Bereich, gab es zunächst die Fragestellung, wie entsprechende Vorgaben bzgl. des Zugriffs auf Ortsinformationen formuliert und durchgesetzt werden können. Das dabei entstandene Grundkonzept lässt sich mit dem Begriff *Privacy Policies* bezeichnen.
Leonhardt und Magee entwickelten eine der ersten Formalisierungen für Zugriffsberechtigungen auf ortsbezogene Daten [114]. Sie definieren dabei sogenannte *location domains*, die geographische Bereiche beschreiben, und *location objects*, die Objekte im System wie z.B. Benutzer beschreiben. Zugriffsrechte werden dann wahlweise ähnlich der Zugriffsmatrizen von Lampson [110] oder der label-basierten Zugriffskontrolle von Bell und La Padula [20] formuliert.
Snekkenes [169] schlägt eine mögliche Architektur zur Vorhaltung und Zugänglichmachung von Privacy Policies vor. Dabei ist vorgesehen, dass die Policies der Nutzer bei sogenannten *Policy Custodians* gespeichert werden. Möchte eine Entität auf die Ortsinformation eines Benutzers zugreifen, so frägt sie beim *Location Provider* an, welcher wiederum über ein *Policy Custodian Directory* den zugehörigen Policy Custodian ausfindig macht und bei diesem die Freigabeerlaubnis für die ursprüngliche Anfrage erbittet. Je nach Bewertung durch den Custodian anhand der nutzereigenen Policy wird die Ortsinformation ausgeliefert oder nicht.
Es gab und gibt immer wieder Bestrebungen, standardisierte Verfahren zur Formulierung von Privacy Policies zu entwickeln. Viele scheitern jedoch an der letztendlichen, weitflächigen Implementierung wie beispielsweise das Platform for Privacy Preferences Project (P3P) [191] des World Wide Web Consortium (W3C) [198].

Bewertung Die Verwendung von Privacy Policies ist im Hinblick auf die in Kapitel 2 skizzierten Anforderungen und Angreifermodelle als komplementär zu anderen notwendigen Methoden zur Durchführung von sicheren und privatsphäreschonenden Proxmitätstests zu sehen. Eine Policy erfordert in diesem Fall grundsätzlich das Vertrauen in den Kommunikationspartner. Natürlich ließe sich eine entsprechende Policy von Benutzerseite aus definieren, die einem zentralen PEC genaue Vorgaben macht, wie mit den übermittelten Ortsdaten zu verfahren ist, wie lange und zu welchem Zweck sie gespeichert werden und an wen sie übermittelt werden dürfen. Eine Garantie besteht dadurch für den Nutzer aber technisch gesehen nicht. Ein neugieriger PEC kann die Daten beliebig verwerten.
Gleichzeitig existiert auch kein Schutz vor gefälschten Ortsangaben und damit manipulierten Proximitätswerten. Auch hier könnte in einer analogen Art und Weise vom PEC oder von einer anderen Partei des Proximitätstests theoretisch eine Art *Security Policy* definiert werden, welche die Übermittlung gefälschter

Ortsangaben untersagt. Technisch lässt sich dies allein mit diesem Mechanismus jedoch nicht sicher stellen.

Sinnvoll eingesetzt werden könnten Privacy Policies unter den betrachteten Rahmenbedingungen vor allem als lokale Beschreibungen der Privatsphäre-Vorgaben. Es können auf einem Endgerät verschiedene Methoden zur Proximitätsbestimmung zur Verfügung stehen, die sich in Merkmalen wie den möglichen Einsatzbereichen, der Genauigkeit und Zuverlässigkeit der Proximitätsschätzung und auch der Menge der offengelegten Informationen unterscheiden. Welche Methode wann zum Einsatz kommen soll, könnte analog der Privacy Policies beschrieben werden.

3.2.2 Position Dummies

Die Grundidee der Position Dummies besteht darin, dass ein Benutzer nicht nur eine Ortsinformation zur Verfügung stellt, sondern zusätzlich mehrere gefälschte (die *Position Dummies*), sodass die wahre Position in dieser Gesamtmenge verschleiert wird [101]. Damit die tatsächliche Position nicht durch weitere Informationen, die evtl. über den Benutzer zur Verfügung stehen, oder durch Verfolgung der Ortsangaben über einen längeren Zeitraum in der Gesamtmenge identifiziert werden kann, sind aufwendigere Mechanismen zur sinnvollen Auswahl der Position Dummies notwendig. Shankar et al. [165] versuchen dazu aus historischen Daten per Clustering geographische Bereiche mit ähnlichen Eigenschaften zur tatsächlichen Position zu finden und verwenden diese als Position Dummies. Zur Verwendung für Navigationssysteme u.ä. wurde von Krumm vorgeschlagen, Informationen über das Straßennetz und probabilistische Modelle des Fahrverhaltens zu verwenden, um geeignete Position Dummies zu finden [106].

Bewertung Das Konzept der Position Dummies ist nur für Anwendungsfälle sinnvoll einsetzbar, in denen es (außer eventuell aus Performanzsicht) keine Rolle spielt, ob eine gewünschte Information mit nur einer echten oder zusätzlich noch mit gefälschten Angaben angefragt wird. Dies ist insbesondere für solche Anwendungen zutreffend, in denen der Benutzer lokal die unnötigen Informationen leicht verwerfen und nur die zur echten Position gehörenden Daten verwerten kann.

Damit ist das Konzept per Definition ungeeignet zum Einsatz in proximitätsbezogenen Diensten. Soll die Proximität zwischen zwei Benutzern festgestellt werden, so ist es nicht sinnvoll, widersprüchliche Informationen bzgl. des Aufenthaltsortes zu verwenden, im Sinne von „ich bin entweder hier oder dort“. Einen Schutz gegen Fälschungen bieten Position Dummies natürlich ebenfalls nicht, ist es doch gerade Teil des Konzepts, falsche Informationen zu liefern.

3.2.3 Mix Zones

Das Konzept der *Mix Zones* wurde ursprünglich von Beresford et al. entwickelt [21]. Das Verfahren basiert darauf, dass geographische Zonen definiert werden, in denen die Benutzer keine Anfragen an den Location Server stellen und bei deren Betreten die Benutzer das Pseudonym, mit dem sie beim Location Server anfragen, wechseln. Vor dem Betreten und nach dem Verlassen einer Mix Zone verwendet ein Benutzer also unterschiedliche Pseudonyme. In der Zeit dazwischen, die zum Überbrücken der Mix Zone notwendig ist, werden keine Anfragen versendet, sodass sich die beobachtbaren Trajektorien aller in der Mix Zone befindlichen Benutzer vermischen. Ein Beobachter von außen kann daher die Pseudonyme, die vor dem Betreten und nach dem Verlassen verwendet werden, einander nicht zuordnen.
Mix Zones wurden unter anderem für Fahrzeug-Netze diskutiert [56, 31], wobei generell gezeigt wurde, dass die Art und Weise, wie die Mix Zones angelegt werden, für deren Effektivität wichtig ist [57]. Für Straßennetze wurde unter anderem mit *MobiMix* ein entsprechender Ansatz vorgestellt [149].

Bewertung Der Ansatz der Mix Zones ist in zweierlei Hinsicht nicht auf das Privatsphärenproblem im Falle von Proximitätserkennungsverfahren anwendbar. Zum einen werden hier die mit Hilfe der Pseudonymwechsel die Identitäten verschleiert, was dem eigentlichen Anwendungsfall der Proximitätserkennung widerspricht. Es soll ja gerade herausgefunden werden, welche Entitäten (also Identitäten) in der Nähe voneinander sind.
Zum anderen bezieht sich die Verschleierung beim Einsatz der Mix Zones auf Trajektorien. Wie in Kapitel 2.4.5 erläutert, ist es zu vermeiden, überhaupt nur eine einzelne Ortsangabe offenzulegen, falls die anfragende Entität sich nicht in der Nähe der angefragten Entität befindet.

3.2.4 Anonymität und Cloaking

Die bereits beschriebenen Mix Zones sind primär für Trajektorien geeignet und sollen den Schutz der Privatsphäre durch temporäre Unterbrechung der Anfragen an den Location Server versuchen sicher zu stellen. Daneben gibt es eine ganze Reihe von Ansätzen, die für eine einzelne ortsbezogene Anfrage an sich versuchen, eine gewisse Anonymität des Anfragenden zu gewährleisten.
Die grundlegende Arbeit in diesem Bereich stammt von Gruteser at al. [80], die das ursprünglich von Sweeney entwickelte Konzept der *k-Anonymität* [177] auf den Einsatzbereich der ortsbezogenen Dienste übertragen haben. Man betrachtet ein Subjekt als k-anonym in Bezug auf eine oder mehrere Eigenschaften, wenn es bzgl. dieser Eigenschaften nicht von mindestens $k-1$ anderen Subjekten unterscheidbar ist. Bezogen auf ortsbasierte Dienste darf also die preisgegebene Ortsinformation eines Nutzers nicht von den Ortsinformationen von mindestens $k-1$ anderen Nutzern unterscheidbar sein. Um diese Eigenschaft

zu erfüllen, muss die zu übermittelnde Ortsangabe bei Bedarf modifiziert werden, was auch mit dem Begriff *Cloaking („Verschleiern")* bezeichnet wird. Dies wird im ursprünglichen Konzept von Gruteser et al. dadurch erreicht, dass die Präzision der Ortsangabe in dem Maße verringert wird, wie es die Situation erfordert, um die k-Anonymitätsbedingung zu erfüllen.

Für viele Dienste bringt eine verringerte Präzision der Ortsangabe eine auf die eine oder andere Weise beeinträchtigte Dienstqualität mit sich. So muss für einfache Nächste-Nachbarn-Suchen („Wo befindet sich die nächste Tankstelle?") entweder ein nicht optimales Ergebnis in Kauf genommen oder eine größere Ergebnisliste vom Dienstanbieter geladen und dann lokal gefiltert werden. Alternativ versucht man einen Kompromiss aus beiden Extremen zu erreichen [137].

Da in der Praxis die Privatsphäre-, d.h. in diesem Fall die Anonymitätsanforderungen unterschiedlicher Nutzer voneinander abweichen können, können mit Hilfe von Nutzerprofilen und einem entsprechend angepassten Anonymisierungsverfahren z.B. minimale Anonymitätsniveaus (d.h. Werte für k) oder die maximal akzeptierte Ungenauigkeit der Ortsinformationen eines Nutzers definiert und umgesetzt werden [63, 64].

Die Effektivität von einfachen, nur auf k-Anonymität basierenden Ansätzen kann durch die Analyse von Daten über einen längeren Zeitraum bzw. von mehreren zeitlich zusammenhängenden ortsbezogenen Anfragen eines Nutzers verringert werden, was wiederum durch intelligentere Cloaking-Strategien ausgeglichen werden kann [132, 179].

Ein weiteres Problem stellt jedoch die einfache Definition von k-Anonymität an sich dar: Sie bezieht die möglicherweise eigentlich zu schützenden Informationen des Nutzers nicht mit ein, wenn diese nicht mit den primär betrachteten Ortsdaten übereinstimmen. Die auf k-Anonymität basierenden Ansätze versuchen also nur, die eigentliche Ortsangabe (die Koordinaten) verschiedener Nutzer k-anonym zu machen. Dabei wird jedoch die Semantik der Orte nicht betrachtet. Ein typisches Beispiel hierfür wäre ein Fußballstadion, in dem sich mehrere Zehntausend Menschen befinden. Selbst bei Wahl eines sehr großen ks kann die verschleierte Ortsangabe immer noch sehr eindeutig auf das Stadion hinweisen, sodass die Semantik des Aufenthaltsortes offen gelegt wird, was in der Regel eine deutlich sensiblere Information als abstrakte Koordinaten darstellt. Eine Lösung kann hierbei die Verwendung der genau aus diesem Grund in generischer Form entwickelte *l-Diversität* [124] auch in ortsbezogenen Diensten sein [15]. Darüberhinaus stehen mit den Konzepten *t-Closeness* [117], *p-Sensitivität* [170] und (k, δ)*-Anonymität* [3] weitere Möglichkeiten zur Verfügung, um abzuleiten, wie stark eine Ortsinformation verschleiert werden muss, um das gewünschte Maß an Anonymität zu gewährleisten.

Unabhängig vom Grad der notwendigen Anonymisierung stellt sich die Frage nach einer möglichen Umsetzung eines solchen Systems, da zur Berechnung des notwendigen Verschleierungsgrads eine mehr oder weniger globale Sicht auf

das Gesamtsystem vorhanden sein muss. Viele der Ansätze verlassen sich hierbei auf die Verwendung einer Trusted Third Party (TTP), welche die Anonymisierung vornimmt bevor die Anfragen an den eigentlichen Location Server weitergeleitet werden. Diese Architektur ist durch den Einbezug eines Dritten, der eine globale Sicht auf die unverfälschten Daten erlangt, in der Praxis grundsätzlich in Frage zu stellen.

Aus diesem Grund hat sich eine Reihe von Ansätzen entwickelt, die versuchen, die entsprechenden Anonymisierungsstrategien in einem verteilten, meist vollständig P2P-basierten System umzusetzen. In PRIVÉ [67] bilden die Nutzer ein P2P-Netz, in dem sich basierend auf den Aufenthaltsorten der Nutzer Cluster bilden. Jeder Cluster besitzt einen *Cluster Head*, der wiederum an einem übergeordneten Cluster beteiligt ist, wodurch sich rekursiv eine Baumstruktur aufbaut. Soll eine ortsbezogene Anfrage gestellt werden, wird innerhalb des Clusters der notwendige Anonymisierungsgrad bestimmt, die Anfrage entsprechend modifiziert und dann von einem zufällig ausgewählten Mitglied des Clusters an den Location Server gesendet. Die Antwort wird vom selben Mitglied entgegen genommen und an den ursprünglichen Sender weitergeleitet. Damit ist kein zentraler Anonymisierungsdienst mehr notwendig. Eine Weiterentwicklung von PRIVÉ stellt MOBIHIDE [66] dar, das sich jedoch hauptsächlich in der Ausgestaltung des P2P-Netzes unterscheidet.

Während die Ansätze PRIVÉ und MOBIHIDE auf einem Overlay-P2P-Netz basieren, d.h. darunter die bestehende Netzinfrastruktur verwenden, gehen Chow et al. einen Schrit weiter und setzen ihren P2P-basierten Cloaking-Ansatz zum Teil auch direkt über mobile AdHoc-Verbindungen um [38]. Alternativ können auch hybride Ansätze verfolgt werden, um nicht die ganze Last des Systems auf die mobilen Endgeräte zu verlagern [200].

Grundsätzlich kann das Verschleiern eines Ortes auch unabhängig von einem genauen Anonymitätsmaß durchgeführt werden. Duckham et al. unterscheiden dabei allgemein die drei Merkmale *Inaccuracy*, *Imprecision* und *Vagueness* [46]. Inaccuracy beschreibt dabei die Korrektheit einer Ortsangabe, d.h. ob die Information faktisch richtig ist. Beispielsweise ist die Aussage „Berlin liegt in Frankreich“ in diesem Sinne nicht korrekt. Diese Aussage ist jedoch im Sinne von (Im)Precision präziser als die wiederum korrekt Aussage „Berlin liegt in Europa“. Ebenfalls präziser ist die Aussage „Berlin liegt in Nordeuropa“, welche jedoch vage ist, da die Definition von „Nordeuropa“ von Fall zu Fall unterschiedlich ausfallen kann.

In ihrem eigenen Ansatz zur Verschleierung von Ortsinformationen beschränken Duckham et al. sich dabei auf die Variierung der Precision, also der Genauigkeit der Ortsangabe [45]. Damit ist der Ansatz sehr ähnlich zu den bereits vorgestellten Arbeiten. Der Unterschied liegt primär darin, dass der Grad der Ungenauigkeit nicht auf Basis einer globalen Sicht, sondern nur anhand lokaler Informationen festgelegt wird. Dies hat den Vorteil, dass keine globalen Informationen und damit keine zentrale Instanz oder verteilte Kooperation

notwendig ist. Der Nachteil ist jedoch eine fehlende Abschätzung, inwieweit die verschleierten Daten tatsächlich einen Schutz der Privatsphäre gewährleisten.

Bewertung Die Ansätze zur Anonymisierung der ortsbezogenen Anfragen, d.h. solche, welche die enthaltenen Ortsinformationen in geeigneter Weise verschleiern, sind auf den ersten Blick nicht passend für eine privatsphäreschonende Proximitätserkennung. Bei dieser spielt ja – wie schon bei den Mix Zones beschrieben – die Identität der betrachteten Entitäten eine wichtige Rolle. Es ist also nicht zielführend zu versuchen, einen Nutzer durch Verschleiern der Ortsangabe „anonymer“ zu machen, wenn auf der anderen Seite die Identität sowieso bekannt ist.
Es gibt jedoch zwei primäre Beispiele, bei denen die erläuterten Techniken durchaus zum Einsatz kommen können. Zum einen ist wie in Kapitel 2.6 beschrieben ein Systemaufbau vorstellbar, bei dem der PEC, also die Komponente, die die Berechnung der Proximität vornimmt, eine unabhängige dritte Partei ist, bei der die interessierten Nutzer dann wiederum Proximitätsinformationen erfragen können. In diesem Aufbau könnten die Benutzer dem Location Server nach einem der obigen Ansätze verschleierte Ortsdaten zur Verfügung stellen, evtl. mit einem Pseudonym versehen, das für den PEC keine Bedeutung besitzt, sodass die Proximitätskomponente auf diesen verschleierten Daten die Nähe der Nutzer untereinander berechnen kann. Dies zieht in den meisten Fällen eine Verschlechterung der Ergebnisqualität nach sich, kann aber für bestimmte Anwendungsfälle trotzdem ausreichend sein.
Zum anderen könnten die weiterführenden Ansätze, die auch die Semantik der Aufenthaltsorte berücksichtigen können, wie z.B. l-Diversität oder t-Closeness, auch im Rahmen von Proximitätserkennungsverfahren angewendet werden. In Kapitel 2.4.5 wurde gefordert, dass die andere am Proximitätstest teilnehmende Partei den Aufenthaltsort des Nutzers aus den übermittelten Daten nicht herausfinden kann, sofern keine Proximität besteht. Diese Anforderung könnte aufgelockert bzw. dahingehend präzisiert werden, dass es der anderen Partei nicht möglich sein soll, die Semantik des Ortes abzuleiten. Beispielsweise soll sie nicht unterscheiden können, ob sich der Nutzer im Krankenhaus oder im benachbarten Kaufhaus befindet. Für manche Anwendungsfälle kann es ausreichend sein, eine Proximität im Kilometerbereich festzustellen, sodass in der Regel eine ausreichende semantische Diversität der möglichen Aufenthaltsorte gegeben ist. Für die in dieser Arbeit jedoch u.a. im Hinblick auf kontextbezogene soziale Netze und Gruppensituationen gesuchte Proximitätserkennung ist eine Ungenauigkeit in dieser Größenordnung nicht mehr akzeptabel.

3.2.5 Koordinaten-Transformation und grid-gestützte Ansätze

Im Hinblick auf ortsbezogene Dienste, die nicht von einzelnen Nutzern individuell sondern von größeren Nutzergruppen kollaborativ genutzt werden, exis-

tiert eine weitere Klasse von privatsphäreschützenden Ansätzen, welche die Ortsinformationen nicht unpräziser machen, sondern nur transformieren. Die Grundidee dabei ist, dass unter den Benutzern eine Transformationsvorschrift bekannt ist, die auf die Ortsinformationen angewandt wird, bevor diese an den Location Server übermittelt werden. Der Location Server erhält damit eine für ihn beliebige Ortsangabe, die erst von den Empfängern, also den anderen Nutzern wieder ausgewertet werden kann.
Einer der ersten Ansätze stammt von Treu et al. [183], die u.a. für den Anwendungsfall des Buddy Trackings ein entsprechendes System konzipiert haben. Dabei werden die Ortsinformationen durch eine distanzerhaltende Transformation modifziert, sodass sogar eine Proximitätserkennung auf den transformierten Daten durchgeführt werden kann. Die Autoren bleiben jedoch vage bzgl. einer konkreteren Umsetzung des vorgeschlagenen Verfahrens. Gutscher et al. skizzieren ein ähnliches System [82], das zudem unterschiedliche Arten von Anfragen (Position Queries, Range Queries, etc.) unterstützt.
Ähnlich gelagert sind grid-gestützte Ansätze wie der von Siksnys et al. [167]. In diesem Fall werden die Positionen der Nutzer auf ein Grid abgebildet und mit einem gemeinsamen Geheimnis verschlüsselt, bevor sie an den Location Server übermittelt werden. Dieser kann die so vorbereiteten Positionen auf Proximität überprüfen, ohne die echten Positionen zu lernen.

Bewertung Die Koordinaten-Transformation, vor allem wenn sie distanzerhaltend ausgelegt ist, und auch die grid-gestützten Ansätze sind zunächst vielversprechende Ideen zur Umsetzung von privatsphäreschonender Proximitätserkennung. Eine Herausforderung liegt jedoch darin, die gemeinsame Transformationsvorschrift bzw. ein gemeinsames Geheimnis unter den Nutzern zu vereinbaren und vor unerlaubtem Zugriff zu sichern. Es handelt sich dabei im Endeffekt um einen gemeinsamen Schlüssel, sodass viele Probleme (und auch Lösungsansätze) analog zum Austausch von Schlüsseln in kryptographischen Systemen zum Tragen kommen. Ein großes Problem besteht in der Anfälligkeit für Brute-Force-Attacken: Ein Teilnehmer kann für relevante Orte die richtig transformierte Repräsentation generieren – selbst wenn er sich nicht an diesem Ort befindet – und diese in den Proximitätstest einschleusen und damit mit gewisser Wahrscheinlichkeit den Ort des anderen Nutzers „erraten“.
Dies ist unabhängig von der Gefährdung der Privatsphäre die größte Schwäche des Ansatzes. Jeder Nutzer kann für sich selbst einen beliebigen Aufenthaltsort vortäuschen. Dies steht den geforderten Eigenschaften des Proximitätserkennungsverfahrens entgegen (vgl. Kapitel 2.4.5).

3.2.6 Private Proximity Testing

Die bisher gezeigten Ansätze zur Behandlung von Privatsphäre-Herausforderungen in ortsbezogenen Diensten versuchen grundsätzlich, die Privatsphäreanforderungen durch Modifikation der bereitgestellten Daten

zu erfüllen. Daneben gibt es eine weitere Klasse von Ansätzen, die die Ortsdaten unverändert lassen und versuchen, mit Hilfe kryptographischer Methoden die Privatsphäre zu schützen. Diese Ansätze basieren grundsätzlich auf der Idee des Private Information Retrieval (PIR) [36, 35, 108, 60]. PIR im Allgemeinen erlaubt es, bestimmte Operationen auf einer verschlüsselten Datenbank durchzuführen, wobei auch die Anfrage so verschlüsselt ist, dass der Empfänger der Anfrage den Inhalt nicht lesen, wohl aber auf der verschlüsselten Datenbank ausführen kann.
Die Idee, das Konzept des PIR zum Schutz der Privatsphäre in ortsbezogenen Diensten einzusetzen, wurde erstmals von Ghinita et al. aufgegriffen [65]. Der vorgestellte Ansatz verhindert die Offenlegung der Ortsinformation und ist per Design nicht anfällig für Angriffe, die Ortsinformationen durch Analyse zeitlich korrelierender Anfragen ableiten wollen. Unterstützt werden approximierte und exakte Nächste-Nachbarn-Anfragen.
Zwei Ansätze befassen sich dediziert mit der Thematik der Proximitätserkennung. Sowohl Narayanan et al. [144] als auch Nielsen et al. [146] schlagen vor, das sogenannte Private Proximity Testing (PPT) in Form von Private Equality Testing (PET) [51, 26, 119] umzusetzen. PET ermöglicht die Prüfung auf Gleichheit zweier Eingaben, ohne dass die Kommunikationspartner die Eingabe des anderen extrahieren können. Die grundsätzliche Idee ist in beiden Arbeiten die Abbildung der geographischen Position auf eine oder mehrere vordefinierte, über den betrachteten Bereich gelegte, Gitterzellen, deren Identifikatoren dann im Rahmen eines oder mehrerer PETs verglichen werden. Bei Gleichheit wird die Proximität erkannt.
Besonders interessant ist, dass Narayanan et al. bereits die Verwendung von Location Tags als Eingabedaten vorschlagen. Diese sollen dann durch einen Private Set Intersection (PSI)-Mechanismus [55, 87, 95, 88] verglichen werden. PSI ermöglicht analog zum PET die Bildung der Schnittmenge zweier Eingabemengen, ohne dass die Kommunikationsparteien die Eingabemenge des jeweils anderen im Klartext erhalten.
Eine große Herausforderung zur Umsetzung der verschiedenen Klassen des PIR ist die praktische Umsetzung, da die Verfahren in der Regel einen ernormen Zusatzaufwand selbst für einfache Operationen mit sich bringen, sodass ein echter Einsatz meist aufgrunf der Effizienz scheitert [168]. Dies ist jedoch ein sehr aktives Forschungsgebiet, in dem deutliche Fortschritte zu erkennen sind, die den tatsächlichen Einsatz von PIR-Verfahren in der Zukunft erlauben können [59].

Bewertung Die vorgestellten Verfahren zum PPT erscheinen in der Theorie äußerst vielversprechend. Die Berechnung von Gleichheit bzw. Ähnlichkeit zweier Eingabedaten, ohne dass die Parteien das jeweils andere Eingabedatum kennenlernen, entspricht sehr genau den Anforderungen an die Privatsphäreeigenschaften des gesuchten Proximitätserkennungsverfahrens. Es erscheint dabei sogar möglich, tatsächliche absolute Ortskoordinaten zu verwenden (bei

Bedarf auch durch Cloaking verfälscht), sodass nahezu beliebige Granularitäten der Proximitätserkennung möglich sind.
Diesem großen Potential bzgl. des Schutzes der Privatsphäre steht zum einen die nach wie vor unzureichende Effizienz der Verfahren gegenüber und zum anderen – was auch in der Zukunft ein generelles Problem bleiben wird – die Möglichkeit, gefälschte Ortsdaten in den Proximitätstest einzuschleusen. Auch dieses Verfahren ist also ein reines Verfahren zum Schutz der Privatsphäre und bietet keine Sicherheit gegen Fälschungen der Ortsangaben der Nutzer.
Bereits bei Narayanan et al. [144] wurde ansatzweise das Potential erkannt, das eine Kombination der PIR-Verfahren mit geeigneten Location Tags bieten kann. Die in den späteren Kapiteln vorgestellten Ansätze könnten grundsätzlich die zweite wichtige Komponente, d.h. die Location Tags, für diese Kombination darstellen.

3.2.7 Zusammenfassung

In diesem Abschnitt wurde eine große Bandbreite an Ansätzen vorgestellt, die auf verschiedene Arten versuchen, die Privatsphäre der Nutzer in ortsbezogenen Diensten allgemein und teilweise in proximitätsbezogenen Diensten im Speziellen in gewissem Maße zu schützen. Die Konzepte aus den Bereichen Privacy Policies, Position Dummies und Mix Zones können im Wesentlichen als ungeeignet für die in dieser Arbeit im Fokus stehende sichere und privatsphäreschonende Proximitätserkennung eingestuft werden. Auch die Mehrzahl der auf Anonymisierung und Cloaking ausgerichteten Verfahren sind im Hinblick auf die gestellten Anforderungen (vgl. Kapitel 2.4.5) nicht einsetzbar.
Grundsätzlich interessant erscheinen die Ansätze zur Koordinaten-Transformation. Läge der Fokus allein auf dem Schutz der Privatsphäre, so bestünde in dieser Richtung Potential, ein geeignetes Verfahren zu finden bzw. zu entwicklen. Da sie jedoch, genauso wie die zuletzt eingeführten Verfahren aus dem Bereich des PIR, keinen Schutz gegen das Einspielen gefälschter Ortsdaten bieten, und damit sowohl indirekt die Privatsphäreanforderungen als auch direkt die geforderte Fälschungssicherheit nicht erfüllen, konnte bisher kein Ansatz ermittelt werden, der die Anforderungen an eine sichere und privatsphäreschonende Proximitätserkennung vollständig erfüllt.
Im folgenden Abschnitt werden nun Ansätze behandelt, die den Fokus auf der anderen primären Anforderung an das gesuchte System haben, nämlich dem Schutz vor gefälschten Ortsdaten.

3.3 Ortsbeweise

Wie in den Anforderungen an das gesuchte Proximitätserkennungsverfahren festgehalten (vgl. Kapitel 2.4.5), ist neben dem Schutz der Privatsphäre vor allem auch der Schutz vor der Verwendung gefälschter Ortsangaben eine primäre Aufgabe. Diese ist im Prinzip sogar als noch wichtiger zu erachten, da

ansonsten über den Umweg des Einspielens gefälschter Ortsangaben auch die Privatsphäre beeinträchtigt werden kann.
Da die Teilnehmer des Proximitätstests auf irgendeine Art und Weise belegen müssen, dass sie sich auch wirklich am vorgegebenen Ort aufhalten, müssen sie eine Art *Ortsbeweis* erbringen. Im Folgenden werden dahingehend drei Klassen von Ansätzen vorgestellt. Die erste Klasse arbeitet dabei auf Basis von Reichweiten-Messungen, die zweite Gruppe basiert im Wesentlichen auf der Idee, dass lokale „Zeugen" die Ortsangabe bestätigen müssen, und die dritte Kategorie nutzt die schon mehrfach erwähnten Location Tags, die auch im weiteren Verlauf der Arbeit eine zentrale Rolle einnehmen.

3.3.1 Timing- und Reichweiten-basierte Ortsbeweise

Mit die ältesten Ansätze zur Erbringung von Ortsbeweisen bzw. auf der Gegenseite zur Verifikation von Ortsbehauptungen basieren auf physikalischen Effekten, insbesondere der Ausbreitungsgeschwindigkeit bzw. Laufzeit verschiedener Signale.
Brands und Chaum versuchten bereits 1994 die Sicherheit von Systemen dadurch zu erhöhen, dass für die erfolgreiche Durchführung eines Protokolls die örtliche Nähe des Durchführenden zu einer gewünschten Entität vorausgesetzt wird [27]. Mittel der Wahl dabei ist die sogenannte *Distance Bounding* Technik, die mit Hilfe einer Laufzeitmessung eines Challenge-Response-Protocols eine obere Grenze für die Entfernung des Kommunikationspartners abschätzt. Im Bereich von mobilen Ad-Hoc-Netzen versuchen Hu et al. durch sogenannte *Geographical Leashes* über die Laufzeit der Funksignale eine maximale Entfernung des Absenders zu verifizieren [90]. Für die Zeitmessung von Funksignalen ist eine sehr genaue Uhr notwendig, die in der Regel auf Standard-Hardware nicht zur Verfügung steht. Aus diesem Grund greifen Sastry et al. [159] bei ihrem *Echo*-Protokoll auf Ultraschallsignale zurück. Während dafür in der Regel eine ausreichend präzise Uhr zur Verfügung steht, scheitert es meist an der nicht vorhandenen Fähigkeit von Standard-Endgeräten, Ultraschall-Signale zu senden bzw. zu empfangen. Corner et al. stellten als eine der ersten einen Ansatz zur von ihnen so genannten *Zero-Interaction Authentication* vor [40]. Die Idee ist, dass sich ein Benutzer nicht durch eine aktive Handlung, z.B. die Eingabe eines Passworts, an seinem Laptop authentifizieren muss, sondern indem er ein mobiles Token bei sich trägt, das über eine kurzreichweitige Funktechnologie die Authentifizierung am Laptop vornimmt. Der Beweis der notwendigen örtlichen Nähe wird hier also durch die beschränkte Reichweite des Funksignals erbracht.

Bewertung Die Ansätze aus der Kategorie der timing- und reichweitenbasierten Ortsbeweise weisen mehrere Nachteile auf. Eine so genaue Zeitmessung, wie sie bei timingbasierten Ansätzen meist nötig ist, ist auf aktueller Standardhardware meist nicht umsetzbar. Sowohl bei der Zeitmessung als auch bei der

physikalischen Reichweite der Funktechnologien (insbesondere, wenn es sich nur um einzelne Signale handelt) kommt es zu größeren Schwankungen, die eine genaue Eingrenzung der örtlichen Nähe erschweren. Für gewisse Anwendungsfälle kann ein solches Verfahren geeignet sein, um einen ausreichenden Ortsbeweis zu erbringen. Für die in dieser Arbeit betrachteten Proximitätsanforderungen scheinen die Ansätze jedoch weniger geeignet.

3.3.2 Kollaborative Ortsbeweise

Ein zweite Klasse zur Erbringung von Ortsbeweisen stellen Verfahren dar, die diese durch die Kollaboration mehrerer Systemteilnehmer erzeugen. Die Grundidee dieser Verfahren besteht darin, dass ein Ortsbeweis dadurch erbracht werden kann, dass sich vor Ort befindende Entitäten eine Bestätigung für die Ortsbehauptung ausstellen, die der Benutzer, der den Ortsbeweis erbringen will, dann dem entsprechenden Empfänger als Beleg mitliefern kann.
Sariou et al. verwenden WLAN-Access-Points (APs) als lokal präsente Entitäten, die für mobile Endgeräte auf Anfrage Ortsbeweise ausstellen [158]. Dazu sendet ein mobiles Endgeräte eine entsprechende Nachricht an den AP, digital signiert um die Authentizität sicher zu stellen. Der AP wiederum antwortet mit einer Nachricht, die den gewünschten Ortsbeweis beinhaltet, wiederum digital signiert, diesmal jedoch vom AP. Mit Hilfe des Datenpakets kann das mobile Endgeräte dem System den aktuellen Aufenthaltsort beweisen.
Der Ansatz von Luo et al. basiert auf der gleichen Grundidee [123]. Auch hier werden Ortsbeweise von modifizierten APs ausgestellt. Es handelt sich jedoch nur um sogenannte *Intermediate Location Proofs*, die über weitere Stationen zu einem finalen Ortsbeweis weiterentwickelt werden. Hierbei wird unter anderem eine *Cheating Detection Authority* durchlaufen, die ungewöhnliche Sprünge in den bewiesenen Orten (also wenn diese unwahrscheinlich weit auseinander liegen) findet und meldet.
Auch Khan et al. verwenden in ihrer prototypischen Umsetzung ihrer *Location Assertion Protocol Architecture* WLAN-fähige Komponenten als *Location Authority*, die die Ortsbeweise ausstellt [100]. Der primäre Beitrag der Arbeit liegt jedoch in der Entwicklung eines Protokolls bzw. genauer Vorschriften, welche Nachrichten welche Informationen enthalten müssen, damit verschiedene Angriffe, wie z.B. auch das spätere Leugnen der Anwesenheit an einem Ort vermieden werden können.
Im sogenannten *Applaus*-System von Zhu et al. werden statt fest installierter WLAN-APs in der Nähe befindliche Bluetooth-fähige Endgeräte zur Ausstellung der Ortsbeweise verwendet. Es ist hier zwar auf der einen Seite nicht notwendig, die Infrastruktur zu modifzieren wie in den WLAN-basierten Ansätzen, auf der anderen Seite müssen jedoch ausreichend viele weitere Teilnehmer des Systems (d.h. „Zeugen“) in der Umgebung sein, damit ein Benutzer von diesen einen Ortsbeweis erfragen kann. Der Vorteil ist wiederum, dass festgelegt werden kann, wie viele Ortsbeweise ein Benutzer „einsammeln“ muss,

damit die Ortsbehauptung akzeptiert wird.
Gambs et al. stellen mit PROPS ein *PRivacy-preserving lOcation Proof System* vor, das verschiedene Privatsphären-Aspekte bei der Generierung von Ortsbeweisen berücksichtigt. So wird z.B. sichergestellt, dass der exakte Ort der Zeugen, welche die Ortsbeweise ausstellen, nicht offengelegt wird. Es ist jedoch auch hier nicht berücksichtigt, dass ein (in der Arbeit sogenannter) *Prover*, also die Entität, die den Ortsbeweis erbringen will, den eigenen Ort nur preisgeben will, wenn sie sich auch am relevanten Ort befindet. Für den betrachteten Anwendungsfall des Systems ist dies auch nicht relevant, für die Einsetzbarkeit als Proximitätserkennungsverfahren im Sinne dieser Arbeit ist der Ansatz damit jedoch nur schwer einsetzbar.

Bewertung Kollaborative Ortsbeweise sind grundsätzlich eine sinnvolle Konstruktion, um die Nähe einer Entität zu einer anderen (in den meisten Fällen hier zu einem Objekt) zu belegen. Nachteil vieler Verfahren ist die Notwendigkeit, dedizierte Infrastruktur auszubringen bzw. die vorhandene zu modifizieren. Andere wiederum benötigen eine ausreichende Dichte an am System beteiligten Entitäten, um in ausreichendem Maße die benötigten „Zeugen“ zur Verfügung zu haben. Für eine Proximitätserkennung im Sinne dieser Arbeit sind sie daher nur bedingt geeignet.
Abstrakt betrachtet könnte man jedoch den später in Kapitel 4 neu vorgestellten Proximitätserkennungsansatz auch teilweise dieser Kategorie zuordnen, da auch dort im Prinzip von umgebenden WLAN-Endgeräten Ortsbeweisähnliche Beiträge geleistet werden. Der neue Ansatz benötigt jedoch keine Mitwirkung der umgebenden Geräte und die von diesen Geräten übermittelten Daten sind auch erst in der Masse „beweiskräftig“, weswegen der Ansatz am ehesten doch der im nächsten Abschnitt vorgestellten Kategorie zugeordnet werden kann: den Location Tags.

3.3.3 Location Tags

Eine weitere Möglichkeit zu Erbringung von Ortsbeweisen stellen die sogenannten Location Tags dar, die in Abschnitt 2.4.6 allgemein eingeführt wurden. Obwohl das Konzept bereits in Arbeiten wie von Qui et al. [153] beschrieben wurde, wurde die Idee der Verwendung von Location Tags zur privatsphäreschonenden Umsetzung von Proximitätserkennung zuerst von Narayanan et al. vorgeschlagen [144]. Wie in Kapitel 2.4.6 beschrieben, handelt es sich bei Location Tags um eine Zusammenstellung physikalischer Gegebenheiten, die einen Ort zu einer bestimmten Zeit charakterisieren und die möglichst die Eigenschaften der Reproduzierbarkeit und Nicht-Vorhersagbarkeit erfüllen sollen. Die Autoren erwähnen verschiedene Datenquellen wie Funksignale (Wireless Local Area Network (WLAN), Bluetooth, GPS, GSM), Audiosignale oder die Anteile bestimmter chemischer Elemente in der Luft als potentielle Kandidaten zur Erstellung von Location Tags. Für WLAN schlagen sie vor, den Datenver-

kehr, primär Broadcast-Nachrichten, in einem Netz zu verwenden. Wie genau die darauf basierenden Location Tags aussehen und in welcher Art und Weise sie verglichen werden sollen, wird jedoch nicht beschrieben. Zudem merken die Autoren an, dass die damit notwendige Rahmenbedingung, dass beide Teilnehmer des Proximitätstests mit dem gleichen WLAN verbunden sein müssen, in der Praxis nicht sinnvoll ist.

Aufbauend auf dem Vorschlag von Narayanan et al. wurden verschiedene Ideen zur Konstruktion von Location Tags entwickelt. Lin et al. verwenden Mobilfunksignale um Location Tags zu erstellen [118]. Sie verwenden dabei die Inhalte des *GSM Paging Channels*, die von allen Geräten innerhalb derselben GSM Location Area (LAC) in gleichem Maße empfangen werden müssten. In der Evaluation stellen sie fest, dass die so konstruierten Location Tags tatsächlich die geforderten Eigenschaften bzgl. Reproduzierbarkeit und Nicht-Vorhersagbarkeit gut erfüllen. Für das in der vorliegenden Arbeit betrachtete Kernszenario der Proximitätserkennung, das sich insbesondere auf sehr nahe, d.h. Gruppensituationen, bezieht, ist der Ansatz aufgrund der Größe der Fläche, die als Proximitätsgebiet gilt, nicht geeignet: Eine LAC kann bis zu $100km^2$ umfassen.

Zheng et al. [203] entwickeln eine zur vorliegenden Arbeit sehr ähnliche Vorstellung davon, wie ein Location Tag sich im besten Fall verhalten bzw. welche Eigenschaften er haben sollte. Sie stellen im Vergleich zu früheren Arbeiten deutlicher in den Mittelpunkt, dass Location Tags keinen Aufschluss über den Ort bzw. dessen absolute Position zulassen sollten, an der sie erstellt wurden, und gleichzeitig nicht fälschbar sein dürfen. Sie schlagen vor, sowohl Steuernachrichten des Long-Term Evolution (LTE)-Mobilfunkstandards als auch die Nachrichten-Header des WLAN-Standards zu verwenden, um die Location Tags zu konstruieren. Gerade im Hinblick auf letztere Datenquelle ist der Ansatz ähnlich zum später in Kapitel 4 vorgeschlagenen, ebenfalls auf WLAN basierenden Ansatz. Ein großer Unterschied liegt doch in der Art und Weise, wie die WLAN-Signale verwendet werden. Das Verfahren von Zheng stützt sich rein auf die eigentlichen Daten des WLAN-Verkehrs. Dies hat zur Folge, dass – wie von den Autoren selbst evaluiert – die Location Tags in einem Umkreis von ca. 30 Metern zu einem positiven Proximitätstestergebnis führen. Das später in dieser Arbeit vorgeschlagene Verfahren nutzt neben den Daten des WLAN-Verkehrs zusätzlich die physikalischen Eigenschaften, d.h. die Signalstärke, um Proximitäten in einem deutlich kleineren Bereich zu erkennen.

Das Verfahren von Mathur et al. arbeitet wiederum nur auf Basis von sehr feinen physikalischen Effekten [133]. Die Autoren versuchen orts- und zeitabhängige kryptographische Schlüssel zu generieren, die ebenso als Location Tags aufgefasst werden können. Sie verwenden dabei Schwankungen in den Amplituden und Phasen verschiedener Funksignale (TV im Frequenzband zwischen 512 und 608 MHz, FM-Radio im Frequenzband zwischen 88 und 108 MHz), die generell unabhängig voneinander verlaufen, in einem Bereich unterhalb der halben Wellenlänge der Signale jedoch korrelieren. Folglich können Geräte,

die sich in diesem Bereich befinden, ähnliche Effekte beobachten. Für die vorgeschlagenen Signalarten liegt die maximale Entfernung bei 1,3 Metern, was auch für den betrachteten primären Anwendungsfall einer Gruppensituation sehr nah erscheint. Verbunden mit der Tatsache, dass die verwendeten Funksignale (TV, FM-Radio) in der Regel mit Standard-Smartphones jedoch nicht benutzt werden können, und die typischerweise verfügbaren Funktechnologien (WLAN, Bluetooth) noch höhere Frequenzen (2,4 GHz bzw. 5 GHz) und damit kürzere Wellenlängen aufweisen, erscheint die Idee zwar in der Theorie sehr interessant, in der Praxis jedoch nur für sehr spezielle Anwendungsfälle einsetzbar.

Bewertung Die Grundidee der Location Tags, also die Verwendung von (physikalischen) Eigenschaften der Umgebung die spezifisch für einen Ort zu einer bestimmten Zeit sind, ist eine der vielversprechendsten im Hinblick auf die geforderten Eigenschaften für ein Proximitätserkennungsverfahren. Die primäre Herausforderung besteht darin, eine geeignete Datenquelle zur Erstellung der Location Tags zu identifizieren: Die Proximitätsbereich muss zum Anwendungsfall passen, die Location Tags dürfen nicht vorhersagbar bzw. berechenbar sein, damit diese nicht gefälscht werden können, und sie sollten nach Möglichkeit keine Informationen über den tatsächlichen Ort, d.h. dessen absolute Position, preisgeben. Location Tags, die alle diese Anforderungen erfüllen, sind damit nicht nur geeignet um Ortsbeweise zu erbringen, sondern haben auch das Potential, dies in einer privatsphäreschonenden Art und Weise zu tun. Die bisher vorgeschlagenen Verfahren erfüllen jedoch die in dieser Arbeit angestrebten Ziele nicht vollständig. Das Konzept der Location Tags an sich ist jedoch sehr sinnvoll und bildet damit auch die Grundlage der in dieser Arbeit vorgeschlagenen Ansätze.

3.3.4 Zusammenfassung

In diesem Abschnitt wurden verschiedene Ansätze zur Umsetzung von Ortsbeweisen vorgestellt. Der Nachteil von Timing- und Reichweiten-basierten Ortsbeweisen liegt ähnlich wie bei kollaborativen Ortsbeweisen darin, dass bestimmte Voraussetzungen wie eine genaue Uhr auf der einen Seite oder eine dedizierte Infrastruktur auf der anderen Seite notwendig sind. Für die in Kapitel 2.4.5 aufgestellten Anforderungen, insbesondere dass das Verfahren unabhängig von externer Infrastruktur und mit Standard-Hardware funktionieren soll, sind diese Eigenschaften problematisch.
Am vielversprechendsten erscheint das Konzept der Location Tags. Diese haben das Potential, alle gestellten Anforderungen zu erfüllen. Die Herausforderung liegt im Finden geeigneter Datenquellen zur Konstruktion der Location Tags, was in den bisher vorgestellten Ansätzen noch nicht vollumfänglich in Hinblick auf die gewünschten Eigenschaften gelungen ist. In der vorliegenden Arbeit werden neue Ansätze vorgestellt, wie sinnvolle Location Tags aufgebaut

werden könnten.

3.4 Gruppenerkennung

Wie in Kapitel 2.4.5 definiert, besteht das primäre Szenario für die in dieser Arbeit betrachtete Proximitätserkennung in der Erkennung von Situationen mit sehr großer örtlicher bzw. kontextueller Nähe, d.h. im Wesentlichen handelt es sich dabei um Gruppensituationen. Um diese zu erkennen kann jedoch statt des bisher grundsätzlich gewählten Umwegs über die Erkennung örtlicher Nähe auch versucht werden, direkt eine Gruppensituation anhand anderer Merkmale zu erkennen, z.B. über die Aktivitäten, welche die beteiligten Personen ausüben.
Ein intuitive Herangehensweise zur Erkennung von Gruppensituationen bzw. genauer gesagt sozialer Interaktionen liegt darin zu versuchen, die Gespräche der Personen untereinander zu erkennen, d.h. wer mit wem spricht. Choudhury et al. verwenden hierzu eine Kombination aus einer Erkennung der Personen in der Nähe über einen speziellen Infrarot-Sensor und der Analyse der Audiodaten, um dann die tatsächlichen Gesprächspartner bzw. -gruppen herauszufiltern [37]. Ihr Ansatz basiert auf einer genauen Analyse des kompletten Audio-Stroms, sodass erhebliche Datenmengen übertragen und miteinander verglichen werden müssen.
Brdiczka et al. verfolgen einen ähnlichen Ansatz, versuchen die Gruppen jedoch nur anhand der Sprechphasen der einzelnen Teilnehmer zu identifizieren [28]. Dazu trägt jeder Teilnehmer ein Anklipp-Mikrofon, mit Hilfe dessen relativ leicht herauszufinden ist, wann die Person spricht. Unter Berücksichtigung der Hypothese, dass Mitglieder einer Gruppe in der Regel einen synchronisierten Ablauf der Sprechphasen (d.h. einer spricht, die anderen hören zu) aufweisen, und dies im Vergleich mit anderen Gruppen nicht der Fall ist, können allein anhand der Sprechphasen bzw. -sequenzen die Gruppen identifiziert werden. Der Ansatz ist interessant, ähnelt er doch von der Grundidee der Verwendung der Sprechaktivitätssequenzen dem später in Kapitel 6 vorgestellten Verfahren. Jedoch wurden in diesem Fall nur Gruppen in einer Gesamtteilnehmermenge von vier Personen identifiziert, was wiederum eine vorherige grundsätzliche Proximitätsbestimmung mit einem anderen Verfahren erfordert.
Zur Analyse des Verhaltens von Menschenmengen bzw. Besucherströmen gibt es einige Ansätze, welche die Gruppenstruktur in einer Gesamtmenge durch Beobachtung von außen mittels Computer-Vision-Techniken versuchen zu erkennen. Ge et al. [62] argumentieren dabei ähnlich wie in der vorliegenden Arbeit, dass sich Menschen innerhalb einer Gruppe ähnlich verhalten bzw. bewegen. Sie verwenden zur Gruppenerkennung dabei zum einen die örtliche Nähe von Personen zueinander, zum anderen die Geschwindigkeit und Richtung, in der sie sich fortbewegen. Diese Information werden aus dem Bildmaterial von stationären Kameras extrahiert. Haritaoglu et al. verwenden ein ähnliches Verfahren, um „Shopping-Gruppen“ innerhalb eines Ladens zu identifizieren

[86]. Nachteil dieser Ansätze neben der nötigen Infrastruktur ist die Tatsache, dass dabei nur abstrakte Gruppen erkannt werden, und mit den einzelnen „Objekten“ keine tatsächlichen Identitäten verbunden sind. Dies ist jedoch notwendig, um das Verfahren im Rahmen der Anwendungsbeispiele wie in Kapitel 2.3.2 skizziert sinnvoll einsetzen zu können.
Marin-Perianu et al. verwenden dedizierte Beschleunigungssensoren und berechnen die Korrelation der rohen Sensordaten verschiedener Teilnehmer untereinander, um diejenigen zu identifizieren, die zusammen unterwegs sind [130]. Durch die schiere Menge an Daten scheint dieses Verfahren jedoch für den Live-Einsatz ungeeignet. In ähnlicher Weise verwenden Gordon et al. die Daten verschiedener Sensoren (z.B. Beschleunigungssensor) eines Android-Gerätes, das von den Teilnehmern am Oberschenkel getragen wird, zur Ableitung der „sozialen Nähe“ der Teilnehmer und darauf aufbauend durch einen Filterschritt der Gruppenzusammengehörigkeiten [79]. Die Sensordaten werden dabei nicht in roher, sondern in einer durch Features repräsentierten Form verwendet. Der grundsätzliche Ansatz ähnelt dem später in Kapitel 6 vorgestellten Verfahren, bewegt sich jedoch auf einem geringeren Aggregationsniveau der Sensordaten und wurde nur für eine kleine Menge an Teilnehmern umgesetzt, deren örtliche Nähe bereits bekannt war.
Kjærgaard et al. benutzen eine Kombination aus WLAN-basierter Positionierung und Abgleich von Daten des Beschleunigungssensors und des Kompasses um Gruppen zu identifzieren [102]. Ihre Argumentation entspricht dabei der auch in dieser Arbeit betrachteten Gegebenheit, dass sich räumlich nah beieinander befindende Entitäten nicht zwingend in einer sozialen Gruppe befinden müssen. Unter Einbezug der Aktivitäten – in diesem Fall repräsentiert durch diverse Eigenschaften der Sensordaten – lässt sich die Zusammengehörigkeit einzelner Personen besser identifizieren. Die Verwendung von WLAN-basierter Positionierung im traditionellen Sinne erfüllt jedoch die in dieser Arbeit gestellten Sicherheits- und Privatsphäreanforderungen an das gesuchte Proximitätserkennungsverfahrens nicht.
Roggen et al. [157] schlagen eine ähnliche Verarbeitungskette vor, wie sie in dieser Arbeit später in Kapitel 6 verwendet wird. Zunächst sollen individuelle Aktivitäten einzelner Personen durch maschinelles Lernen erkannt werden. Anschließend können diese Aktivitäten dann mit anderen Personen verglichen werden, um ähnliche und damit in der gleichen Gruppe befindliche Personen identifizieren zu können. Der Fokus liegt hier jedoch auf einzelnen Aktivitäten und deren spezieller Ausprägung und weniger – wie später in der vorliegenden Arbeit – auf Aktivitätssequenzen. Zudem wird wiederum nur in einer kleinen Gruppe evaluiert und es fehlt eine Aussage, wie (gut) aus einer großen Gesamtpopulation Gruppen extrahiert werden können.

Bewertung Die gezeigten Verfahren zur Gruppenerkennung erscheinen zum Teil vielversprechend bzgl. einer generellen Proximitätserkennung. Vor allem bzgl. des in der vorliegenden Arbeit diskutierten primären Szenarios sind die

Grenzen der beiden Themengebiete fließend. Auszuschließen sind zunächst Verfahren, die von einer dedizierten Infrastruktur abhängig sind und/oder Gruppen nur abstrakt erkennen, ohne die entsprechenden Identitäten damit in Verbindung bringen zu können. Zudem betrachten die meisten Verfahren die Privatsphäreimplikationen des Austausches der benötigten Daten nicht. Die Verfahren auf Basis von Bewegungs- bzw. Aktivitätsdaten gehen in eine ähnliche Richtung wie das in Kapitel 6 der vorliegenden Arbeit vorgestellte Verfahren. Jedoch beziehen sich diese durchweg auf sehr eng abgegrenzte Bereiche, in denen eine räumliche Proximität der Teilnehmer bereits durch den Grundaufbau gegeben ist, und nur noch aus einer begrenzten Menge Gruppen feiner unterschieden werden müssen. Zur Reduzierung der Gesamtpopulation auf diese Initialmenge ist also wiederum ein (anderes) Proximitätserkennungsverfahren notwendig.

3.5 Zusammenfassung

In diesem Kapitel wurde ein Überblick über verwandte Arbeiten gegeben, die durch verschiedene Zielsetzungen bzw. Eigenschaften eine Lösung bzw. Teillösung der in Kapitel 2.4.5 definierten Problemstellung mit den dort definierten Rahmenbedingungen und Anforderungen sein könnten. Eine vollumfänglich passende Lösung wurde nicht gefunden.
Für den weiteren Verlauf der Arbeit sind die folgenden Erkenntnisse weiterhin von Interesse:

- Die Ansätze des PPT (vgl. Abschnitt 3.2.6) sind eine mögliche, wenn auch rechenaufwendige Teillösung der Privatsphäre-Anforderung, dass ein Teilnehmer des Proximitätstests den tatsächlichen Standort – in diesem Fall sogar die kompletten Eingabedaten – des anderen Teilnehmers nicht ableiten können soll, wenn sich die Teilnehmer nicht am gleichen Ort zur gleichen Zeit befinden. Eine Teillösung ist dies deshalb, weil die Ansätze ein Raten der Position in der Hoffnung auf eine zufällig gefundene Proximität nicht verhindern können. In Verbindung mit einem anderen Verfahren, das das Einschleusen falscher Ortsangaben verhindert, könnte eine vollumfängliche Lösung konstruiert werden.

- Im Hinblick darauf erscheinen die Ansätze auf Basis von Location Tags vielversprechend (vgl. Abschnitt 3.3.3). Die Herausforderung liegt dabei nach wie vor im Finden eines geeigneten Ausgangsmaterials für die Location Tags, sowie einer passenden Art und Weise, wie diese Tags erstellt und miteinander verglichen werden können. Diesbzgl. werden im weiteren Verlauf zwei Ansätze vorgestellt.

- Wie vor allem bei den Ansätzen zur Gruppenerkennung ersichtlich (vgl. Abschnitt 3.4), ist es in der verwandten Literatur eine weit verbreitete Annahme, dass sich die Gruppenzugehörigkeit einer Person und damit

die Proximität zu anderen Personen in einer Ähnlichkeit des Verhaltens bzw. der Aktivitäten dieser Personen äußert. Diese Annahme stellt auch (teilweise) die Grundlage zweier im weiteren Verlauf der Arbeit vorgestellter Verfahren dar.

Nachdem nun ein Überblick über verwandte Arbeiten gegeben wurde, wird in den folgenden drei Hauptkapiteln jeweils ein Verfahren vorgestellt, das die gestellten Anforderungen an das gesuchte Proximitätserkennungsverfahren besser erfüllt. Das erste Verfahren basiert dabei primär auf räumlicher Proximität, während das zweite implizit in Teilen und das dritte schließlich rein auf der Erkennung der Aktivitäten der Benutzer beruht.

4 Proximitätserkennung mit Hilfe von WLAN-Management-Frames

In diesem Kapitel wird das erste der in der vorliegenden Arbeit neu entwickelten Verfahren zur sicheren und privatsphäreschonenden Proximitätserkennung vorgestellt. Das Verfahren beruht auf der Erstellung von nicht fälschbaren und nicht vorhersagbaren Location Tags auf Basis von WLAN-Signalen.
Die Grundidee besteht darin, dass sich die Menge der an einem Ort befindlichen Personen sowie deren Positionen ständig und in nicht zuverlässig vorhersagbarer Weise verändern. Diese Zusammensetzung von Personen an einem Ort kann also theoretisch als quasi-zufällige Datenquelle zur Erstellung von ortsspezifischen Fingerabdrücken verwendet werden. Es stellt sich nur die Frage, wie die an einem Ort befindlichen Personen automatisiert „gemessen“ werden können. Da für die Zukunft davon ausgegangen werden kann, dass die meisten Menschen ein oder mehrere WLAN-fähige Endgeräte mit sich führen [122], können diese als Repräsentation der Personen betrachtet werden.
Um die in der Umgebung befindlichen WLAN-Endgräte zu ermitteln, wird eine Eigenheit des WLAN-Protokolls ausgenutzt. Dieses sieht zum Auffinden bekannter APs den Broadcast bestimmter Management-Frames vor, die im Klartext gesendet werden und damit von in Reichweite befindlichen Geräten ausgewertet werden können. Zeichnet man also diese Management-Frames aller umliegenden Geräte auf, kann aus den jeweiligen Sender-Adressen und der beobachteten Signalstärke ein für den Ort spezifischer Fingerabdruck erstellt werden. Dieser Location Tag kann zum Vergleich einer anderen Entität zur Verfügung gestellt werden. Er enthält keine Ortsinformation an sich, es kann also daraus kein absoluter Ort bestimmt werden, wenn man sich nicht selbst am selben Ort aufhält. Andererseits kann im Falle tatsächlicher räumlicher Nähe zwischen zwei Kommunikationspartnern ein entsprechendes Ergebnis berechnet werden.
Im Folgenden werden nun zunächst die Grundidee (Abschnitt 4.2) sowie die für den Ansatz relevanten Grundlagen erläutert (Abschnitt 4.3). Nach einem kurzen Überblick über weitere verwandte Arbeiten (Abschnitt 4.4) wird das Konzept im Detail vorgestellt (Abschnitt 4.5) und theoretische Eigenschaften veranschaulicht (4.6). Der Ansatz wurde in mehreren Experimenten umfangreich evaluiert. Die Ergebnisse und Erkenntnisse daraus werden im Abschnitt 4.7 dargestellt. Abschließend erfolgt eine zusammenfassende Diskussion des

Ansatzes (Abschnitt 4.8).

4.1 Vorveröffentlichungen

Die Kerninhalte dieses Kapitels wurden vom Autor bereits in [129] publiziert. Wie in Kapitel 1.3 dargestellt, stammen die im Paper und im Folgenden präsentierten Inhalte bzgl. der Idee, des Konzepts, der theoretischen Eigenschaften und der Evaluation vom Autor der vorliegenden Arbeit. Ebenfalls im Paper bereits enthalten sind die Abbildungen 4.2, 4.3, 4.4, 4.5, 4.6, 4.7 und 4.8.

4.2 Motivation und Grundidee

In den Kapiteln 2.4.6 und 3.3.3 wurde bereits erläutert, dass für die geforderten Eigeschaften an das Proximitätserkennungsverfahren das Konzept der Location Tags allgemein als sehr vielversprechend anzusehen ist. Die Grundidee besteht darin, auf irgendeine Art und Weise mess- oder erfassbare Eigenschaften eines Ortes zur Beschreibung desselbigen zu verwenden. Es stellt sich dabei nur die Frage, welche Eigenschaften eines Ortes tatsächlich geeignet sind, um sinnvoll einsetzbare Location Tags zu erzeugen. Im Hinblick auf die Anforderungen an das Proximitätserkennungsverfahren wurden in Kapitel 2.4.6 insbesondere die beiden Charakteristiken „Reproduzierbarkeit“ und „Nicht-Vorhersagbarkeit“ als essentielle Merkmale sinnvoller Location Tags identifiziert.
Um diese beiden Eigenschaften zu erhalten, wird zu Erzeugung der Location Tags eine Datenquelle benötigt, die zum einen für einen kurzen Zeitraum an einem Ort relativ stabile Werte liefert und sich zum anderen aber möglichst zufällig verhält. Als „kurzer Zeitraum“ kann dabei abstrakt die Zeitspanne gesehen werden, in der sich der Kontext eines Menschen i.d.R. nur wenig ändern kann. Im Hinblick auf die Fortbewegungsgeschwindigkeit eines Menschen und seiner Umgebung können hier als grober Richtwert Zeiträume im Bereich von Sekunden bis hin zu mehreren Minuten angesehen werden.
Eine Datenquelle, die diese Eigenschaften erfüllt, sind die Menschen selbst bzw. die Zusammensetzung der Personen an einem bestimmten Ort zu einem bestimmten Zeitpunkt. Betrachtet man öffentliche Bereiche, wie beispielweise einen großen zentralen Platz in einer Stadt, so befinden sich dort viele Personen, die sich unabhängig voneinander an diesem Ort aufhalten und umher bewegen. Die Zusammensetzung und räumliche Konstellation dieser Menschen verändert sich in oben angesprochenem „kurzen Zeitraum“ nur wenig, sie ändert sich jedoch über einen längeren Zeitraum gesehen fortwährend, und es ist extrem unwahrscheinlich, dass sich die gleiche Zusammensetzung in der gleichen räumlichen Anordnung an diesem Ort – oder auch an einem beliebigen anderen – wiederholt. Könnte man also messen, welche Personen sich um einen Nutzer herum in welcher relativen Postion befinden, so könnte man dies zur Erzeugung eines Location Tags verwenden.

Dies ist die Grundidee des im Folgenden präsentierten Verfahrens. Dabei besteht die Idee zudem darin, dass die Personen, die sich in der Umgebung eines Nutzers befinden, tatsächlich „gemessen“ werden können, und zwar anhand der WLAN-Signale, die ihre mitgeführten mobilen Endgeräte aussenden. Da die meisten Menschen mittlerweile ein WLAN-fähiges mobiles Endgerät (z.B. Smartphone) besitzen und bei sich tragen [122, 171], können diese tatsächlich als Repräsentation der eigentlichen Personen angesehen werden.

4.3 Grundlagen des WLAN-Protokolls

Beim im weiteren Verlauf neu vorgestellten Ansatz zur Proximitätserkennung werden bestimmte Protokollbestandteile des IEEE 802.11 (WLAN) Standards [94] ausgenutzt, um in der Nähe befindliche WLAN-Endgeräte zu identifizieren. Der WLAN-Standard unterscheidet drei verschiedene Klassen von Datenrahmen:

- *Control Frames*: Dienen zum Management des Zugriffs auf das gemeinsame Medium und unterstützen den Übertragungsprozess von Datenrahmen.
- *Data Frames*: Transportieren die eigentlichen Nutzdaten.
- *Management Frames*: Dienen zum Austausch von Informationen zum Verbindungsaufbau und zur Verbindungssteuerung.

Besonders interessant für den neuen Ansatz sind die Management Frames, die am Prozess zum Auffinden und Beitreten von WLANs beteiligt sind. Der WLAN-Standard sieht für diesen Prozess zwei grundsätzliche Varianten vor, einen *passiven* Beobachtungsprozess und einen *aktiven* Scanprozess.
Bei der passiven Variante senden WLAN-APs in regelmäßigen Abständen (ca. alle 100ms) *Beacons* aus, um die eigene Präsenz mitzuteilen. WLAN-Endgeräte, die sich im passiven Scan-Modus befinden, beobachten diese Beacons und reagieren, wenn sich ein gewünschtes Netz in Reichweite befindet. Die Beacons werden von den APs jedoch nur auf dem Kanal ausgesendet, den der AP aktuell benutzt, sodass die Endgeräte auf verschiedenen Kanälen auf Beacons lauschen müssen, um alle vorhandenen APs identifizieren zu können. Im Gegensatz zur passiven Variante werden beim aktiven Scanprozess nicht zuerst die APs sondern die Endgeräte aktiv. Diese senden sogenannte *Probe Requests* aus, um das Vorhandensein von ihnen bekannten APs zu überprüfen. Probe Requests werden von den Endgeräten auf allen WLAN-Kanälen ausgesendet, sodass der entsprechende AP den Probe Request auf jeden Fall auf seinem Kanal empfängt. Die Probe-Request-Frames enthalten die MAC-Adresse der Netzwerkschnittstelle des Absenders und – wenn es sich um einen gerichteten Probe Request handelt – die Service Set Identification (SSID) des gesuchten Netzes. Im Falle eines gerichteten Probe Requests antwortet nur der

entsprechende AP (sofern in Reichweite), bei ungerichteten Probe Requests, d.h. bei einem leeren SSID-Feld im Datenrahmen, antworten alle APs in Reichweite.
Für mobile Endgeräte hat sich die aktive Scan-Variante als sinnvoller erwiesen, da sie im Vergleich zum passiven Verfahren die vorhandenen APs schneller identifizieren kann, bei geringerem Energieverbrauch [112]. Aktuelle Endgeräte führen den Scan mindestens alle zwei Minuten durch, selbst wenn sie bereits mit einem WLAN verbunden sind [24]. In der Praxis werden primär gerichtete Probe Requests eingesetzt, d.h. die mobilen Endgeräte senden Probe Requests für jedes Netz (d.h. jede SSID), das sie in ihrer Liste von bekannten WLANs gespeichert haben.
Da Beacons und Probe Requests als Management Frames zum Auffinden von gesuchten WLANs dienen und damit zum Einsatz kommen, bevor eine konkrete Verbindung zu einem WLAN besteht, werden diese Datenrahmen im Klartext verschickt. Es ist damit allen WLAN-Geräten in Empfangsreichweite möglich, die Frames aufzuzeichnen und auszulesen.

Für den neu entwickelten Ansatz zur Proximitätserkennung bilden die Probe Requests die Grundlage zur Erstellung der Location Tags. Aus den oben beschriebenen Protokoll-Vorgaben ergeben sich folgende Eigenschaften, die die Probe Requests dafür zur idealen Datenquelle machen:

- Im Gegensatz zu den Beacons, die von den meist stationären APs versendet werden, werden Probe Requests von den meist mobilen WLAN-Endgeräten wie Smartphones, Tablets oder Laptops ausgesandt, sodass sich die Zusammensetzung der sendenden Geräte an einem Ort ständig ändert.
- Da es sich um Management-Frames zum Auffinden von APs handelt, senden alle mobilen Endgeräte mit aktiviertem WLAN Probe Requests aus, unabhängig davon, ob aktiv Daten übertragen werden sollen und ob das Gerät bereits mit einem WLAN verbunden ist.
- Probe Requests werden oft genug versendet, sodass auch bei geringerer Endgeräte-Dichte ausreichend Daten erzeugt werden.
- Probe Requests werden unverschlüsselt versendet und können mit aktueller Standard-Hardware aufgezeichnet und ausgewertet werden.

4.4 Verwandte Arbeiten aus dem Bereich der WLAN-Positionierung

Neben den bereits in Kapitel 3.3.3 vorgestellten verwandten Arbeiten im Bereich der Location Tags, sind für das neu vorgeschlagene Verfahren grundsätzlich auch Arbeiten aus dem Bereich der Positionierung mit Hilfe von WLAN

relevant. Verfahren, die auf Basis von WLAN-Signalen die Position eines Nutzers bestimmen, existieren sowohl für Innen- [14, 109, 5] als auch Außenbereiche [139, 147].

Die Verfahren bestehen i.d.R. aus zwei Teilen, einer *Offline-Phase* zur Erzeugung einer WLAN-Fingerprint-Datenbank und einer *Online-Phase*, in der die tatsächliche Positionierung durchgeführt werden kann. In der Offline-Phase werden in dem Gebiet, in dem das Postionierungssystem später angewendet werden soll, an möglichst vielen Standorten WLAN-Fingerprints – also im Prinzip Location Tags – aufgenommen. Diese bestehen in der Regel aus einer Liste von WLAN-APs, die an diesem Standort erreichbar sind, sowie der jeweiligen Signalstärke, mit der sie beobachtet wurden. Jeder Fingerprint wird zusammen mit den Ortskoordinaten, an denen er aufgenommen wurde, in einer Datenbank gespeichert.

In der Online-Phase wird nun von einem Endgerät, das die eigene Position herausfinden will, ebenfalls ein aktueller Fingerprint aufgenommen, und dieser dann mit den Einträgen der Datenbank verglichen. Je nach System kommt z.B. ein *k-Nächste-Nachbarn*-Verfahren zum Einsatz, um die aktuelle Position des Endgeräts zu schätzen.

Ein wichtiges Ziel dieser Verfahren ist allgemein, eine möglichst robuste, d.h. über längere Zeit stabile, Fingerprint-Datenbank zu erzeugen. Fingerprints sollen sich also über die Zeit möglichst wenig ändern, sodass die Vergleiche mit neuen Online-Fingerprints der aufgezeichneten Orte erfolgreich verlaufen und die Datenbank möglichst selten aktualisiert werden muss. Obwohl diese Systeme technologisch auf ähnliche Grundelemente wie das im Folgenden neu vorgestellte Verfahren aufbauen, ist die Zielsetzung und das Vorgehen also *konträr* zu den gewünschten Eigenschaften des Proximitätserkennungsverfahrens. Bei letzterem soll gerade keine zeitliche Stabilität vorhanden sein, um die Nicht-Vorhersagbarkeit zu gewährleisten, und damit die Anforderungen bzgl. Sicherheit und Privatsphäre der Location Tags zu erfüllen.

Neben der Arbeiten im Bereich der (Indoor-)Positionierung, die sich wie gesehen primär auf stationäre WLAN-Endgeräte stützen, gibt es einige weitere verwandte Arbeiten, die – ähnlich wie der später präsentierte neue Ansatz – die ausgesandten Signale mobiler WLAN-Endgeräte für diverse Zwecke nutzen bzw. zweckentfremden. Beispielsweise verwenden Chon et al. die beobachtbaren WLAN-Signale mobiler Endgeräte, um die Mobilität innerhalb von Städten zu messen [34]. Auf ähnliche Weise zeigen Schauer et al. eine feingranularere Besucherstrommessung innerhalb von Gebäuden [160]. Barbera et al. analysieren die Inhalte ausgesandter Probe Requests, um sogar persönliche Eigenschaften der Teilnehmer einer öffentlichen Veranstaltung, wie z.B. die von ihnen gesprochene Sprache, abzuleiten [17] und Weppner et al. messen – in diesem Fall jedoch auf Basis von Bluetooth – die Dichte von Menschenmengen anhand der sichtbaren mobilen Endgeräte [192].

Alle diese Verfahren lassen das Potential der Nutzung der „Mitteilungsfreudigkeit“ mobiler Endgeräte zur Umsetzung ganz anderer Dienste erkennen als

mit der Funktechnologie eigentlich beabsichtigt. Im Folgenden wird ein darauf basierendes Konzept zur sicheren und privatsphäreschonenden Proximitätserkennung vorgestellt.

4.5 Konzept zur Proximitätserkennung mit Hilfe von WLAN-Management-Frames

Die Grundidee dieses neuen Ansatzes zur Proximitätserkennung besteht darin, die Menge der WLAN-Endgeräte – die zumeist ein Alias für die sie besitzende Person sind[1] – an einem bestimmten Ort zu einer bestimmten Zeit als Grundlage zu Erstellung nicht fälschbarer und unvorhersagbarer Location Tags zu verwenden. Die WLAN-Endgeräte werden dabei durch die von ihnen ausgesandten Probe Requests identifiziert. Aufgrund der Verwendung von Probe Requests wurden die damit konstruierten Location Tags *ProbeTags* getauft.

In Abbildung 4.1 ist das grundsätzliche Szenario und der grundsätzliche Ablauf exemplarisch dargestellt. Die beiden Benutzer Alice und Bob befinden sich in der Nähe voneinander und sind umgeben von weiteren Personen, die Endgeräte mit aktivem WLAN bei sich tragen. Die MAC-Adressen der WLAN-Schnittstellen der umgebenden Endgeräte sind dabei vereinfacht mit AA:AA, BB:BB, CC:CC und DD:DD bezeichnet. Alice und Bob befinden sich jeweils in der WLAN-Reichweite unterschiedlicher, teilweise gemeinsamer umgebender Personen, visualisiert durch Kreise. Dabei kann Alice die Signale von AA:AA, BB:BB und CC:CC beobachten, Bob dagegen von BB:BB, CC:CC und DD:DD. Mit Hilfe dieser Signale können die beiden einen Proximitätstest durchführen, wie im Folgenden beschrieben.

Der Ablauf des Proximitätstests gliedert sich in zwei Phasen, die Konstruktion der ProbeTags und den Vergleich dieser. Diese werden im Folgenden erläutert.

4.5.1 Konstruktion der ProbeTags

Als Grundlage für die ProbeTags zeichnet jeder Teilnehmer des Proximitätstests für einen vorher definierten Zeitraum der Länge Δt sämtliche Probe Requests auf, die am aktuellen Standort verschickt werden. Alternativ können die Probe Requests auch kontinuierlich aufgezeichnet werden und ein entsprechendes Zeitfenster der Länge Δt kann bei Bedarf extrahiert werden. Geeignete Werte sollten sich im Bereich von Sekunden bis zu wenigen Minuten bewegen, da die Antwortzeit des Verfahrens (bis eine Aussage über Proximität getroffen werden kann) stark von Δt abhängt (der Einfluss dieses Parameters wird später genauer evaluiert).

[1] Im weiteren Verlauf ist technisch das WLAN-Endgerät gemeint, auch wenn zum einfacheren Verständnis von der Person selbst gesprochen wird.

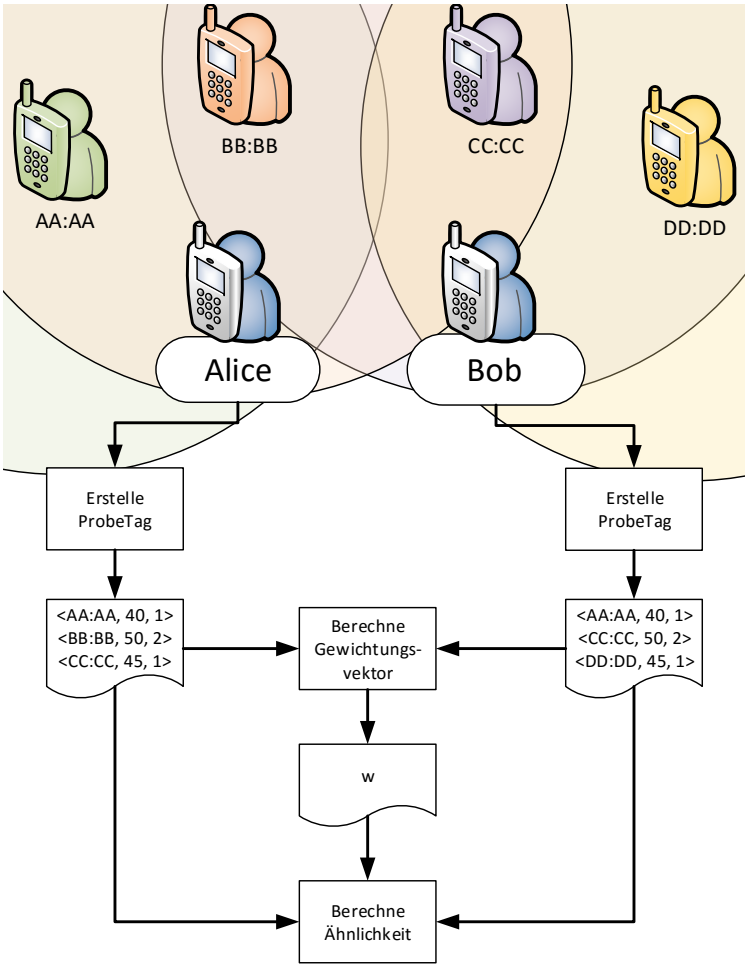


Abbildung 4.1: Beispielszenario und Ablauf des ProbeTag-Verfahrens. Alice und Bob möchten die Proximität bestimmen. Beide empfangen Signale von bestimmten WLAN-Endgeräten aus der Umgebung.

Für jeden Probe Request wird die MAC-Adresse des Absenders, die beobachtete Signalstärke und der lokale Zeitstempel der Beobachtung gespeichert. Der Signalstärke-Wert muss je nach Chipsatz evtl. normalisiert bzw. vorverarbeitet werden. Im Folgenden wird von einem Wertebereich zwischen 0 (niedrigste Signalstärke) und 100 (höchste Signalstärke) ausgegangen. Die aufgezeichneten Probe Requests des gewählten Zeitraums werden dann in einen einzigen ProbeTag aggregiert. Ein ProbeTag ist eine Liste von Tupeln der Form

$$\langle \text{MAC}, \text{RSSI}, \text{NumObservations} \rangle$$

wobei MAC die MAC-Adresse des Absenders ist, RSSI die *durchschnittliche* Signalstärke aller empfangenen Probe Requests von diesem Absender im aktuellen Zeitfenster und NumObservations die Anzahl der empfangenen Probe

Requests von diesem Absender. Folglich existiert für jede MAC-Adresse, von der im aktuellen Zeitfenster ein Probe Request gesendet (und aufgezeichnet) wurde, genau ein Eintrag im ProbeTag.

Für das Szenario aus Abbildung 4.1 sollen für die folgenden Berechnungen beispielhaft folgende Beobachtungen gemacht worden sein: Alice beobachtet AA:AA einmal mit Signalstärke 40, BB:BB zweimal mit durchschnittlicher Signalstärke 50 und CC:CC einmal mit Signalstärke 45. Bob zeichnet ebenfalls Probe Requests auf, einmal von BB:BB mit Signalstärke 40, zweimal von CC:CC mit durchschnittlicher Signalstärke 50 und einmal von DD:DD mit Signalstärke 45. Es ergeben sich also die ProbeTags

$$a = \begin{pmatrix} \langle \text{AA:AA}, 40, 1 \rangle \\ \langle \text{BB:BB}, 50, 2 \rangle \\ \langle \text{CC:CC}, 45, 1 \rangle \end{pmatrix} \qquad b = \begin{pmatrix} \langle \text{BB:BB}, 40, 1 \rangle \\ \langle \text{CC:CC}, 50, 2 \rangle \\ \langle \text{DD:DD}, 45, 1 \rangle \end{pmatrix}$$

Die ProbeTags werden zwischen den Teilnehmern ausgetauscht. Dies geschieht wie im Systemmodell (vgl. Kapitel 2.6) beschrieben über einen zentralen PEC oder direkt P2P.

4.5.2 Vergleich von ProbeTags

Die ProbeTags zweier Teilnehmer, für die ein Proximitätstest durchgeführt werden soll, müssen nun verglichen werden, um eine Bewertung der Ähnlichkeit der beiden ProbeTags und damit der räumlichen Nähe der beiden Teilnehmer zu erhalten. Wie in Kapitel 2.4.1 beschrieben, ist dazu ist eine geeignete Distanz- bzw. Ähnlichkeitsfunktion notwendig.

Jaccard-Koeffizient Eine Möglichkeit, zwei ProbeTags zu vergleichen, besteht zunächst darin, allein die beiden Mengen der beobachteten Sender-MAC-Adressen zu vergleichen. Stimmen die Mengen gut überein, ist anzunehmen, dass die beiden Teilnehmer sich am selben Ort aufhalten. Als Kennzahl für die Ähnlichkeit von Mengen eignet sich der Jaccard-Koeffizient [**?**]. Für n Mengen $S_1, \ldots, S_n$ ist er definiert als

$$J(S_1, S_2, \ldots, S_n) = \frac{|S_1 \cap S_2 \cap \cdots \cap S_n|}{|S_1 \cup S_2 \cup \cdots \cup S_n|} \tag{4.1}$$

Angewandt auf das Szenario aus Abbildung 4.1 ergibt sich für Alice und Bob bei 2 übereinstimmenden MAC-Adressen und insgesamt 4 MAC-Adressen in der Vereinigungsmenge ein Proximitätswert von 0.5.

Um genauere Ergebnisse zu erhalten, erscheint es sinnvoll, die Informationen über die beobachteten Signalstärken ebenfalls in den Vergleich mit einfließen zu

lassen. Man erstellt dafür aus den ProbeTags a und b Signalstärke-Vektoren $\vec{a}$ und $\vec{b}$, indem man für jede beobachtete MAC-Adresse eine Vektorkomponente hinzufügt und mit der beobachteten Signalstärke belegt. MAC-Adressen, die nur in einem der beiden ProbeTags enthalten sind, werden beim anderen aufgefüllt und 0 gesetzt. Wichtig ist, dass die Reihenfolge der Komponenten (d.h. der MAC-Adressen, denen sie entsprechen) bei beiden Signalstärke-Vektoren identisch ist.

Für die ProbeTags a und b aus dem beschriebenen Szenario aus Abbildung 4.1 ergeben sich folgende Signalstärke-Vektoren $\vec{a}$ und $\vec{b}$:

$$\vec{a} = \begin{pmatrix} 40 \\ 50 \\ 45 \\ 0 \end{pmatrix} \qquad \vec{b} = \begin{pmatrix} 0 \\ 40 \\ 50 \\ 45 \end{pmatrix}$$

Für den Vergleich dieser Vektoren ist die am häufigsten verwendete Metrik in verwandten Gebieten wie dem WLAN-Fingerprinting die euklidische Distanz [181]. Eigene Vorexperimente und aktuelle Untersuchungen [181] zeigen jedoch, dass die Verwendung der euklidischen Distanz oft nicht die sinnvollste Variante darstellt. Insbesondere im gegebenen Anwendungsfall entstehen zwei Herausforderungen:

- Die Menge der möglichen MAC-Adressen, die beobachtet werden können, ist in der Praxis nicht vorher bekannt. Dadurch ist es unmöglich, die Signalstärke-Vektoren übergreifend über mehrere Proximitätstests in der selben Form zu konstruieren. Die Menge und Reihenfolge der Vektorkomponenten variiert, was einen Vergleich von berechneten Ähnlichkeitswerten über mehrere Tests hinweg erschwert.

- Die beobachteten Signalstärken können unbeabsichtigt durch das verwendete Endgerät bzw. dessen Trageposition am Körper beeinflusst werden. Unterschiedliche Endgeräte weisen zudem unterschiedliche Empfangsleistungen auf, sodass selbst physikalisch gleiche Signalstärken zu unterschiedlichen Messungen auf den Geräten führen können. Eine Normalisierung erfordert meist eine umfangreichere Kalibrierung jedes einzelnen Gerätes, die zudem immer wieder aktualisiert werden muss, je nachdem ob das Gerät beispielsweise in einer Tasche getragen wird oder nicht. Daher ist die Verwendung von Distanzmaßen, die von den absoluten Vektorbeträgen beeinflusst werden, meist problematisch.

Kosinus-Ähnlichkeit Ein mögliches Ähnlichkeitsmaß, dass von den aufgeführten Problemen weniger stark beeinflusst wird, ist die Kosinus-Ähnlichkeit.

Für zwei Vektoren $\vec{a}$ und $\vec{b}$ ist sie definiert als:

$$\cos(\vec{a},\vec{b}) = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\|\vec{b}\|} = \frac{\sum_{i=1}^{n}\vec{a}_i\cdot\vec{b}_i}{\sqrt{\sum_{i=1}^{n}(\vec{a}_i)^2}\cdot\sqrt{\sum_{i=1}^{n}(\vec{b}_i)^2}} \tag{4.2}$$

Es wird also der Kosinus des Winkels zwischen den beiden Vektoren berechnet. Dieses Distanzmaß hat den Vorteil, dass der Winkel zwischen den Vektoren nicht abhängig von deren Betrag ist und dieser damit keinen Einfluss auf das Ergebnis hat.
In der Theorie liegen die Werte der Kosinusähnlichkeit im Intervall $[-1, 1]$, wobei ein Wert von 0 durch zwei orthogonale, also im Hinblick auf den gegebenen Anwendungsfall komplett verschiedene Vektoren verursacht wird und die Werte 1 bzw. -1 für genau gleich bzw. genau entgegengesetzt gerichtete Vektoren steht. In der Praxis treten im betrachteten Anwendungsfall keine Vektorkomponenten mit im Vergleich zum anderen Signalstärke-Vektor entgegengesetztem Vorzeichen auf, sodass der Wertebereich der Kosinusähnlichkeit auf das Intervall $[0, 1]$ beschränkt bleibt. Damit entspricht sie den in Kapitel 2.4.1 beschriebenen Eckpunkten.

Für das Beispiel aus Abbildung 4.1 ergibt sich eine Kosinusähnlichkeit der beiden Signalstärkevektoren $\vec{a}$ und $\vec{b}$ von 0.69.

Neben der Menge der MAC-Adressen sowie den beobachteten Signalstärken steht eine dritte Information zur Verfügung, die für die Berechnung die Ähnlichkeit der ProbeTags verwendet werden kann: die Anzahl der Beobachtungen jeder MAC-Adresse. Für den praktischen Einsatz des Systems kann diese Information ein wichtiger Bestandteil des Vergleichs sein. Wenn ein Teilnehmer des Proximitätstests im betrachteten Zeitfenster mehrere Nachrichten von einer bestimmten MAC-Adresse beobachten konnte, dann ist die Wahrscheinlichkeit höher, dass auch der andere Teilnehmer diese MAC-Adresse mindestens einmal beobachtet haben muss, sofern sich beide in der Nähe voneinander befinden. Wurde eine MAC-Adresse nur einmal registriert, so ist die Wahrscheinlichkeit, dass der andere Teilnehmer gar keine Beobachtung machen konnte, höher und weniger aussagekräftig. Das vollständige Fehlen einer MAC-Adresse hat jedoch großen Einfluss auf die Richtung des Signalstärke-Vektors und damit das Ergebnis der Kosinus-Ähnlichkeit. Es ist also sinnvoll, den Einfluss der Komponenten des Signalstärke-Vektors anhand der Anzahl der Beobachtungen der jeweils zugehörigen MAC-Adresse zu gewichten.

Gewichtete Kosinus-Ähnlichkeit Die Kosinus-Ähnlichkeit kann in eine gewichtete Form überführt werden, indem ein zusätzlicher Gewichtungsvektor $\vec{w}$ als Eingabeparameter verwendet. Sind $\vec{\omega_a}$ und $\vec{\omega_b}$ Vektoren, die die Anzahl der Beobachtungen der zur jeweiligen Komponente gehörenden MAC-Adresse in ProbeTag a bzw. b beschreiben, und sei λ ein grundsätzlicher Faktor zur

Bestimmung der Intensität der Gewichtung, dann ist der Gewichtungsvektor $\vec{w}$ definiert als

$$\vec{w} = \begin{pmatrix} \lambda \max(\omega_{a1}, \omega_{b1}) \\ \lambda \max(\omega_{a2}, \omega_{b2}) \\ \vdots \\ \lambda \max(\omega_{an}, \omega_{bn}) \end{pmatrix} \tag{4.3}$$

Es wird also die maximale Anzahl an Beobachtungen zur Gewichtung herangezogen und dann (optional) noch durch den Faktor λ modifiziert. Der so konstruierte Gewichtungsvektor fließt dann in folgender Form in die Berechnung der gewichteten Kosinus-Ähnlichkeit $\cos_w$ ein:

$$\cos_w(\vec{a}, \vec{b}, \vec{w}) = \frac{\sum_{i=1}^{n}(\vec{w}_i(\vec{a}_i \cdot \vec{b}_i))}{\sqrt{\sum_{i=1}^{n}(\vec{w}_i\vec{a}_i^2)} \cdot \sqrt{\sum_{i=1}^{n}(\vec{w}_i\vec{b}_i^2)}} \tag{4.4}$$

Im Beispielszenario ergibt sich folgender Gewichtungsvektor:

$$\vec{w} = \begin{pmatrix} \lambda \cdot 1 \\ \lambda \cdot 2 \\ \lambda \cdot 2 \\ \lambda \cdot 1 \end{pmatrix}$$

Für $\lambda = 1$ ergibt sich als Proximitätswert 0.81. Die Tatsache, dass sowohl Alice als auch Bob jeweils die häufiger beobachteten MAC-Adressen (BB:BB und CC:CC) mindestens einmal gesehen haben, führt also zu einem höheren Ähnlichkeitswert.

4.6 Theoretische Eigenschaften des ProbeTag-Verfahrens

Für den ProbeTag-Ansatz ergibt sich theoretisch ein charakteristischer Grenzwert, der die maximale physikalische Distanz beschreibt, in der noch eine Ähnlichkeit > 0 zweier ProbeTags festgestellt werden kann. Damit eine Ähnlichkeit > 0 entsteht, müssen die beiden ProbeTags mindestens einen gemeinsamen Absender enthalten. Die maximale Distanz $\delta_{\max}$, in der dies möglich ist, ergibt sich aus den physikalischen Eigenschaften der WLAN-Signale und der teilnehmenden Entitäten. Sie kann theoretisch durch folgende Formel bestimmt werden:

$$\delta_{\max} = 2r + v_u * w + v_s * w \tag{4.5}$$

Dabei ist r die maximale Reichweite von WLAN-Signalen, v_u die Geschwindkeit der teilnehmenden Entitäten, v_s die Geschwindigkeit der Signal-Quelle (d.h. des einzigen gemeinsamen Senders) und w die Länge des betrachteten Zeitfensters.

Nimmt man für von mobilen Endgeräten ausgesendete WLAN-Signale eine maximale Reichweite von 100 Metern an und dass sich sowohl die Nutzer als auch die Signalquellen mit Schrittgeschwindigkeit, d.h. beispielwiese $5\frac{m}{s}$, bewegen, so erhält man bei einem Zeitfenster von 30 Sekunden eine maximale Distanz, in der noch eine Ähnlichkeit festgestellt werden könnte, von 500 Metern.

4.7 Evaluation des ProbeTag-Verfahrens

Im Folgenden wird der vorgestellte Ansatz zur Proximitätserkennung in mehreren Szenarien evaluiert. Da sich rein aus den theoretischen Eigenschaften bereits ergibt, dass eine grundsätzliche Proximitätserkennung für eine Distanz von maximal etwa 500 Metern möglich ist, sollen vor allem die folgenden beiden Fragen geklärt werden: i) Wie verhalten sich die ProbeTag-Ähnlichkeiten in direkter Nähe bzw. unterschiedlichen Distanzen, und ii) lassen sich ProbeTags, die in direkter Nähe zur gleichen Zeit aufgenommen wurden, gut genug von wiedereingespielten ProbeTags aus der Vergangenheit unterscheiden, um die Sicherheits- und Privatsphäre-Anforderungen zu erfüllen.

4.7.1 Grundsätzlicher Versuchsaufbau und verwendete Komponenten

Für die Datenaufzeichnung der Evaluation wurden vier mobile Endgeräte mit der WLAN-Logging-Software versehen. Die Geräte waren Smartphones des Herstellers Samsung vom Typ Galaxy S II i9100 [**?**]. Um den WLAN-Verkehr in der Umgebung mitlesen zu können, müssen die WLAN-Schnittstellen der Endgeräte in den sogenannten *Monitor Modus* versetzt werden. Dies ist aktuell mit den Standard-Firmwares der bekannten Betriebssysteme (iOS, Android) noch nicht möglich, weswegen die alternative Firmware CyanogenMod 10 [**?**] installiert wurde. Mit Hilfe der Software des bcmon-Projekts [**?**] kann die WLAN-Schnittstelle sehr einfach in den Monitor Modus versetzt werden. Anschließend kann mit Hilfe von Standard-Linux-Werkzeugen wie tcpdump der gesamte WLAN-Verkehr überwacht und aufgezeichnet werden.
Zur Erstellung der Datensätze wurden alle beobachteten WLAN-Management-Frames zusammen mit dem Beobachtungszeitpunkt gespeichert. Für die Auswertung wurden dann aus den Aufzeichnungen die Probe Requests extrahiert.

Neben den WLAN-Daten wurden mit Hilfe einer Logging-Applikation die GPS-Koordinaten der Geräte zusätzlich aufgezeichnet, um für bestimmte Testreihen eine zusätzliche örtliche Referenzinformation zur Verfügung zu haben.

4.7.2 Direkte Proximität

Die beiden ersten durchgeführten Versuchsreihen dienten dazu, das Verhalten des vorgeschlagenen Ansatzes unter nahezu Labor-Bedingungen grundsätzlich zu testen. Dazu wurden zwei Testgeräte direkt nebeneinander auf einem Tisch abgelegt. Bzgl. der Konstellation ist dieses Szenario der beste Fall, der unter realen Bedingungen, also mit echten Endgeräten, möglich ist: Die Geräte befinden sich fast exakt am selben Ort, sie werden nicht bewegt und es existieren keine Hindernisse zwischen ihnen, die den Empfang von Funksignalen beeinflussen können.
Das Ziel dieser Versuche war zum einen zu verifizieren, dass grundsätzlich eine Ähnlichkeit zwischen den ProbeTags zweier so nah beieinander liegender Geräte besteht, und zum anderen herauszufinden, wie stabil die berechneten Ähnlichkeitswerte über die Zeit sind. Da sich am Szenario, d.h. an den Positionen und insbesondere dem physischen Abstand der Testgeräte über den gesamten Zeitraum nichts ändert, ist eine niedrige Schwankung der Ähnlichkeitswerte wünschenswert.
Es wurden Versuchsreihen in zwei unterschiedlichen Szenarien durchgeführt. Im ersten Fall lagen die Testgeräte während normaler Arbeitszeiten auf einem Schreibtisch in einem Einzelbüro, d.h. es gab nur wenige WLAN-Signale von mobilen Endgeräten aus der direkten Umgebung. Das Büro grenzt an zwei Seiten an weitere Büros, an der dritten an einen Flur und an der vierten Seite befindet sich ein Fenster zum Außenbereich vor dem Haupteinang des Gebäudes. Es befinden sich wenig Signalquellen im gleichen Raum, jedoch können aus den Nachbarräumen und insbesondere vom Flur und auch von den Außenbereichen Signale von vorbeilaufenden Personen empfangen werden. Bei einer Fenstergröße von 30 Sekunden die zu einem ProbeTag zusammengefasst werden, konnten in diesem Szenario im Schnitt nur neun verschiedene Signalquellen pro ProbeTag aufgezeichnet werden.
Das zweite Szenario war im Gegensatz dazu eine deutlich belebtere Umgebung. Die Testgeräte wurden in diesem Fall an einem warmen Sommerabend auf einem Tisch in einem öffentlichen Biergarten abgelegt. Der Gastbereich um die Testgeräte herum war sehr gut gefüllt, sodass am Tisch, an den Nebentischen und den Wegen dazwischen eine Vielzahl von Personen und damit Signalquellen vorhanden waren. Dies spiegelt sich auch in den aufgezeichneten Daten wider. Ebenfalls bei einer Fenstergröße von 30 Sekunden konnten in diesem Fall im Durchschnitt 107 Signalquellen pro ProbeTag aufgezeichnet werden.
Im Büro-Szenario zeigten sich dabei die in Abbildung 4.2 dargestellten Ergebnisse. In der Abbildung sind die drei in Kapitel 4.5.2 vorgeschlagenen Ähnlichkeitsmaße getrennt voneinander eingetragen. An der Abszisse ist dabei die Zeit

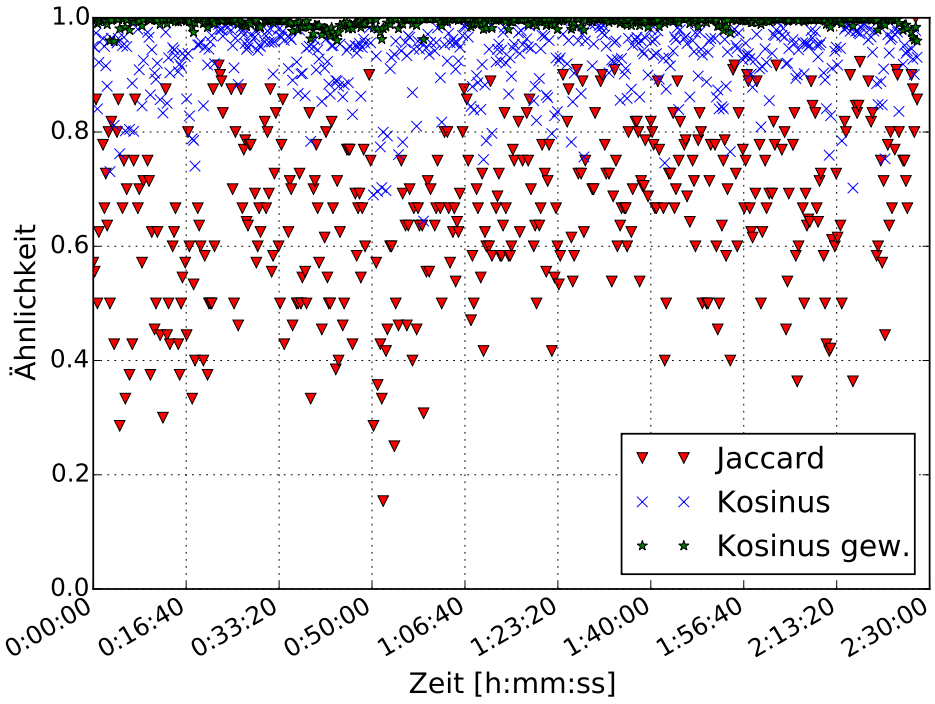


Abbildung 4.2: ProbeTag-Ähnlichkeiten unter Verwendung von Jaccard-, Kosinus- und gewichteter Kosinusähnlichkeit bei direkt nebeneinanderliegenden Testgeräten in einem Büro-Szenario.

des Experiments angegeben, an der Ordinate der berechnete Ähnlichkeitswert. Die Ähnlichkeitswerte des Jaccard-Koeffizienten (rote Dreiecke) sind äußerst variabel und erstrecken sich nahezu auf den gesamten Wertebereich der Funktion. Damit wird zwar grundsätzlich eine Ähnlichkeit erkannt, weitergehende Aussagen lassen sich damit allerdings nicht treffen. Auch ist fraglich, ob damit eine Unterscheidung von wiedereingespielten ProbeTags möglich ist.

Der Verlauf der Ähnlichkeitswerte bei Benutzung der normalen Kosinusdistanz (blaue Kreuze) ist deutlich robuster. Die Werte bewegen sich nurmehr zwischen 0.6 und 1.0, d.h. es existiert ein deutlich größerer Wertebereichs-Puffer in dem wiedereingespielte ProbeTags von validen ProbeTags unterschieden werden können.

Am besten schneidet jedoch die vorgeschlagene Berechnung der gewichteten Kosinusdistanz (grüne Sterne) ab. Hierbei wurde der Gewichtungsfaktor $\lambda = 1$ verwendet. Die Werte weichen über den gesamten Zeitraum nur geringfügig von der maximalen Ähnlichkeit ab. Dies ist sehr nahe am optimalen Ergebnis. Die Werte sind dem tatsächlichen Szenario entsprechend stabil und zudem sehr hoch, sodass ein genügend großer Wertebereich bleibt, in dem Szenarien mit größeren Distanzen sowie wiedereingespielte ProbeTags verortet werden können, ohne die Unterscheidbarkeit zu gefährden.

Die Ergebnisse des zweiten Szenarios sind in Abbildung 4.3 visualisiert. Auf den ersten Blick ist zu erkennen, dass sich sowohl der Jaccard-Koeffizient als auch die normale Kosinus-Distanz deutlich robuster verhalten und weniger streuen. Eine größere Anzahl an Signalquellen führt also zu einer geringeren Varianz der Ähnlichkeitswerte. Auch wenn in dieser Testreihe selbst die Verwendung des Jaccard-Koeffizienten möglich erscheint, bleibt die Kosinus-Ähnlichkeit bzgl. Streuung und Maximalwert überlegen. Die gewichtete Kosinus-Ähnlichkeit zeigt in diesem Szenario eine minimal größere Varianz als in der weniger belebten Büroumgebung, stellt nach wie vor jedoch das mit Abstand beste Ähnlichkeitsmaß der drei Varianten dar.

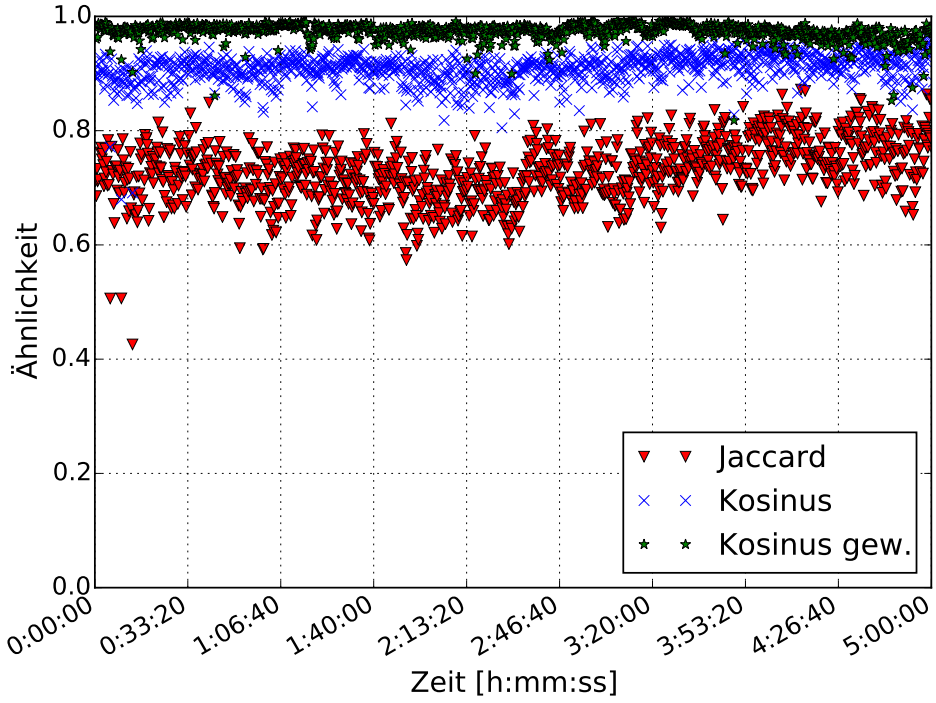


Abbildung 4.3: ProbeTag-Ähnlichkeiten unter Verwendung von Jaccard-, Kosinus- und gewichteter Kosinusähnlichkeit bei direkt nebeneinanderliegenden Testgeräten in einer belebten, öffentlichen Umgebung.

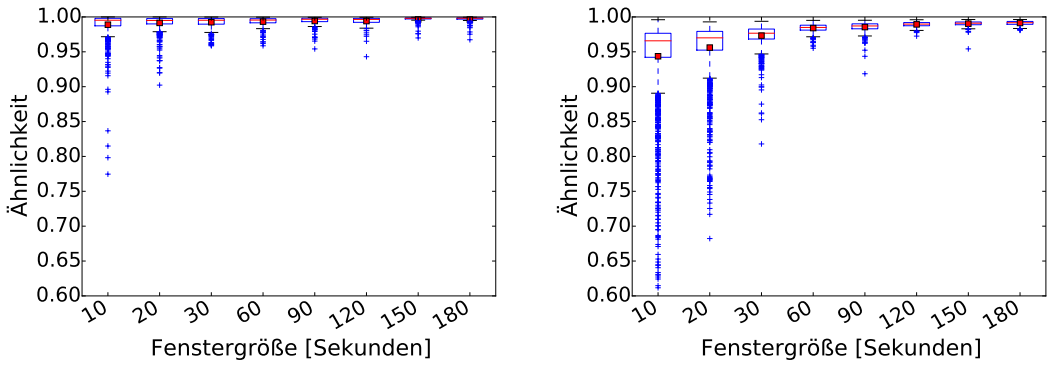
Berücksichtigt man diese beiden grundsätzlichen Testergebnisse, so lassen sich folgende Erkenntnisse ableiten:

- Mehr Signalquellen führen zu robusteren Ergebnissen.
- Bei ausreichend vielen Signalquellen können auch einfachere Ähnlichkeitsmaße wie der Jaccard-Koeffizient sinnvolle Ergebnisse liefern.

- In beiden relativ unterschiedlichen Szenarien zeigt die vorgeschlagene Berechnung einer gewichteten Kosinus-Ähnlichkeit die robustesten Ergebnisse. Zudem unterscheiden sich die Durchschnittswerte bei diesem Ähnlichkeitsmaß zwischen den beiden Szenarien nur unwesentlich, was für eine eventuelle Definition von global gültigen Richtwerten eine wichtige Voraussetzung ist.

4.7.3 Fenstergrößen

Im vorherigen Kapitel wurde gezeigt, dass die gewichtete Kosinus-Ähnlichkeit für die beiden bisher vorgestellten Szenarien sehr gute Ergebnisse liefert. Wie in Kapitel 4.5.1 beschrieben, gibt es neben dem verwendeten Ähnlichkeitsmaß einen weiteren Einflussfaktor auf die Ähnlichkeitsberechnung: die verwendete Fenstergröße. Eine geeignete Größe für das Fenster, aus dem dann ein ProbeTag aggregiert wird, hängt von mehreren Gesichtspunkten ab. Die bisherigen Ergebnisse zeigen, dass eine größere Menge an Signalquellen, d.h. Daten, zu robusteren Ähnlichkeitswerten führen. Die Menge der Signalquellen lässt sich auch durch die Fenstergröße beeinflussen. Werden die ProbeTags über größere Intervalle gebildet, erhöht sich die durchschnittliche Anzahl an Signalquellen. Andererseits führen größere Fenster, wie in Kapitel 4.6 erläutert, direkt auch zu einer größeren maximalen physischen Distanz, für die eine Ähnlichkeit theoretisch erkannt werden könnte. Zudem kann die Antwortzeit des Gesamtsystems ansteigen, da evtl. auf das Ende eines vollständigen Intervalls gewartet werden muss.
In Abbildung 4.4 ist eine Auswertung des Einflusses verschiedener Fenstergrößen auf die durchschnittlichen Ähnlichkeitswerte und deren Varianz dargestellt. Es wurde sowohl der Datensatz aus dem Büro-Szenario (Abbildung 4.4a) als auch der aus dem Biergarten-Szenario ausgewertet (Abbildung 4.4b).



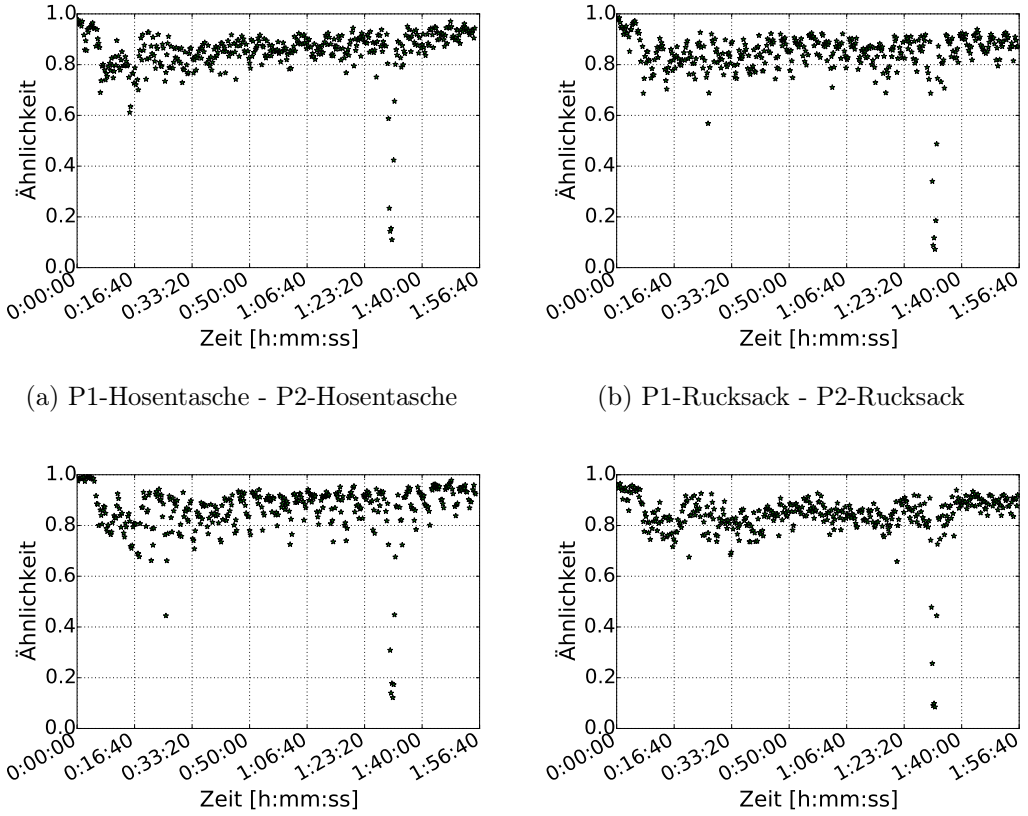
(a) Büro-Szenario (b) Biergarten-Szenario

Abbildung 4.4: Ähnlichkeitswerte der gewichteten Kosinus-Ähnlichkeit in zwei Szenarien bei unterschiedlichen Fenstergrößen.

An der Abzisse finden sich hierbei jeweils die verschiedenen Fenstergrößen in der Einheit Sekunden, an der Ordinate wiederum die Ähnlichkeitswerte. Die Darstellung ist in Form eines Boxplots erfolgt, der für jede Messreihe das arithmetische Mittel, den Median, das untere und obere Quartil, sowie das 2%-Quantil und das 98%-Quantil enthält. Ähnlichkeitswerte, die nicht in den 2%-98%-Quantilsbereich fallen, werden einzeln im Plot dargestellt.
Die gewichtete Kosinus-Ähnlichkeit zeigt sich selbst bei kleinen Fenstergrößen relativ robust. Die Mittelwerte liegen nah beim maximalen Ähnlichkeitswert und auch die Varianz ist bereits bei kurzen Intervallen gering. Problematisch ist nur das Auftreten einiger Ausreißer, die je nach Systemkonfiguration zu Fehlaussagen führen könnten. Ab einer Fenstergröße von 30 Sekunden liegen jedoch in beiden Szenarien 98% der Fenster bei einem Ähnlichkeitswert von 94% oder mehr. Die Verbesserungen durch noch größere Fenster fallen insbesondere im Büro-Szenario nur gering aus, weswegen eine Fenstergröße von 30 Sekunden als sinnvoller Wert angenommen werden kann. Dieser ist im Hinblick auf Reaktionszeit und maximaler Reichweite ebenfalls als geeignet anzusehen.

4.7.4 Proximität bei öffentlicher Veranstaltung

Während in den ersten Versuchsreihen das Verhalten des Systems unter quasi-idealen Bedingungen untersucht wurde, wird im folgenden ein Experiment unter realen Bedingungen erläutert. In diesem Szenario wurden zwei Testpersonen mit jeweils zwei Testgeräten ausgestattet. Jede Person trug eines dieser Geräte in der Hosentasche, das andere im Rucksack. Um eine möglichst realistische Umgebung zu untersuchen, nahmen die beiden Testpersonen gemeinsam zwei Stunden an einer öffentlichen Veranstaltung teil. Sie sollten dabei möglichst die ganze Zeit „zusammen“ unterwegs sein, was für die beiden definiert wurde als „in Gesprächsreichweite, ca. 1 bis 5 Meter“.
Die Datenaufzeichnung wurde dabei bei Erreichen des Festival-Geländes gestartet. Die beiden Testpersonen bewegten sich zunächst ca. 20 Minuten im Freien über das Gelände. Anschließend betraten sie gemeinsam einen großen Innenbereich, in dem sie sich für die nächsten ca. 1,5 Stunden aufhielten. Sie saßen dabei zeitweise an einem Tisch, bewegten sich zwischendurch aber auch an andere Orte im Innenbereich. Die Vorgabe bzgl. der maximalen Entfernung konnte fast durchweg eingehalten werden.
Dieses Experiment unterscheidet sich von den vorherigen Versuchsreihen stark. In diesem deutlich realistischeren Szenario liegen die Testgeräte nicht stationär direkt nebeneinander, sondern werden von echten Testpersonen am Körper mitgeführt. Die tatsächliche Entfernung zwischen den Testgeräten ist daher größer und variiert stärker, insbesondere weil bei einer gut besuchten öffentlichen Veranstaltung die Testpersonen immer wieder zum Ausweichen u.ä. Aktionen gezwungen werden. Durch die Tageposition in der Hosentasche bzw. im Rucksack wird zudem der Empfang der Funksignale durch Körper und Objekte in der Umgebung beeinflusst.



(a) P1-Hosentasche - P2-Hosentasche

(b) P1-Rucksack - P2-Rucksack

(c) P1-Hosentasche - P2-Rucksack

(d) P1-Rucksack - P2-Hosentasche

Abbildung 4.5: Gewichtete Kosinus-Ähnlichkeit der ProbeTags über die Zeit, für zwei Personen (P1, P2), die zusammen eine öffentliche Veranstaltung besuchen und jeweils zwei Testgeräte bei sich tragen.

Die Ergebnisse des Versuchs sind in Abbildung 4.5 dargestellt. In den jeweiligen Plots ist auf der Abszisse wiederum der zeitliche Verlauf des Experiments angetragen und auf der Ordinate der errechnete Ähnlichkeitswert mit der gewichteten Kosinus-Ähnlichkeit. Die einzelnen Abbildungen beziehen sich auf die Kombinationen der verschiedenen Endgeräte des Testnutzer:

- Das Testgerät in der Hosentasche von Testnutzer 1 mit dem Testgerät in der Hosentasche von Testnutzer 2 (Abbildung 4.5a)
- Das Testgerät im Rucksack von Testnutzer 1 mit dem Testgerät im Rucksack von Testnutzer 2 (Abbildung 4.5b)
- Das Testgerät in der Hosentasche von Testnutzer 1 mit dem Testgerät im Rucksack von Testnutzer 2 (Abbildung 4.5c)
- Das Testgerät im Rucksack von Testnutzer 1 mit dem Testgerät in der

Hosentasche von Testnutzer 2 (Abbildung 4.5d)

Es lässt sich gut erkennen, dass die unterschiedlichen Trageposition nur wenig Einfluss auf die grundsätzlichen Ähnlichkeitswerte haben. Hier kommt der bereits erklärte Vorteil der Kosinus-Ähnlichkeit bzgl. der guten Robustheit gegenüber allgemeiner Dämpfung des Signals zum Tragen.
Im Allgemeinen zeigt sich im Vergleich zu den vorherigen Experimenten im Büro und Biergarten eine stärkere Streuung der Ähnlichkeitswerte um einen niedrigeren Mittelwert. Dies ist jedoch sehr gut durch das Szenario erklärbar, in dem die Testgeräte auch in größerer und vor allem in gewissen Grenzen variabler Entfernung positioniert waren. Auffällig ist in allen vier Graphen ein Minimum bei ca. 1 Stunde und 30 Minuten. Dieses ist korrekt, da sich zu diesem Zeitpunkt eine der Testpersonen kurz weiter von der anderen entfernt hat, um die Toilette aufzusuchen. Im Ähnlichkeitsverlauf wurde die nicht mehr vorhandene Proximität erkannt.

4.7.5 Ähnlichkeitsmaße

Nachdem nun drei verschiedene Szenarien eingeführt wurden, wird im Folgenden nochmal zusammenfassend die Güte der drei in Kapitel 4.5.2 vorgeschlagenen Ähnlichkeitsmaße untersucht. In Abbildung 4.6 sind erneut in Boxplots die Verteilungen der Ähnlichkeitswerte der einzelnen Maße in den drei Szenarien dargestellt. Auf der Abszisse sind die jeweiligen Ähnlichkeitsmaße aufgeführt, auf der Ordinate der Ähnlichkeitswert.
Für die beiden Szenarien im Büro und im Biergarten sind die Darstellungen aggregierte Varianten des zeitlichen Verlaufs aus den Abbildungen 4.2 und 4.3. Hinzu kommt jeweils das dritte Szenario, das öffentliche Festival mit den echten Testpersonen.
Beim Jaccard-Koeffizient beobachtet man in allen drei Szenarien eine große Streuung sowie teilweise relativ niedrige Mittelwerte. Die Kosinus-Ähnlichkeit verhält sich in beiderlei Hinsicht deutlich besser. Am besten sind jedoch die Ergebnisse der gewichteten Kosinus-Ähnlichkeit, die in allen drei Szenarien die geringste Streuung der Ähnlichkeitswerte und den höchsten Mittelwert liefert. Für alle weiteren Auswertungen wird daher die gewichtete Kosinus-Ähnlichkeit als Ähnlichkeitsmaß verwendet. Der Ähnlichkeitswert des so festgelegten Verfahrens wird auch als *Proximitätswert* bezeichnet.

Auf Basis der bisher gezeigten Versuchsreihen lässt sich folgern, dass eine grundsätzliche Proximitätserkennung mit Hilfe des vorgeschlagenen Ansatzes der ProbeTags möglich ist. „Grundsätzlich“ bedeutet dabei, dass das Verfahren bei tatsächlicher örtlicher Nähe im Bereich von 0 bis 5 Metern konstant hohe Proximitätswerte zwischen 1.0 und 0.7 zurückliefert. Auf der anderen Seite ist durch die theoretischen Überlegungen (siehe Kapitel 4.6) eine maximale Distanz bei der überhaupt Proximitätswerte > 0 auftreten können gegeben. Um das Verfahren noch genauer bewerten und einordnen zu können, sind zwei

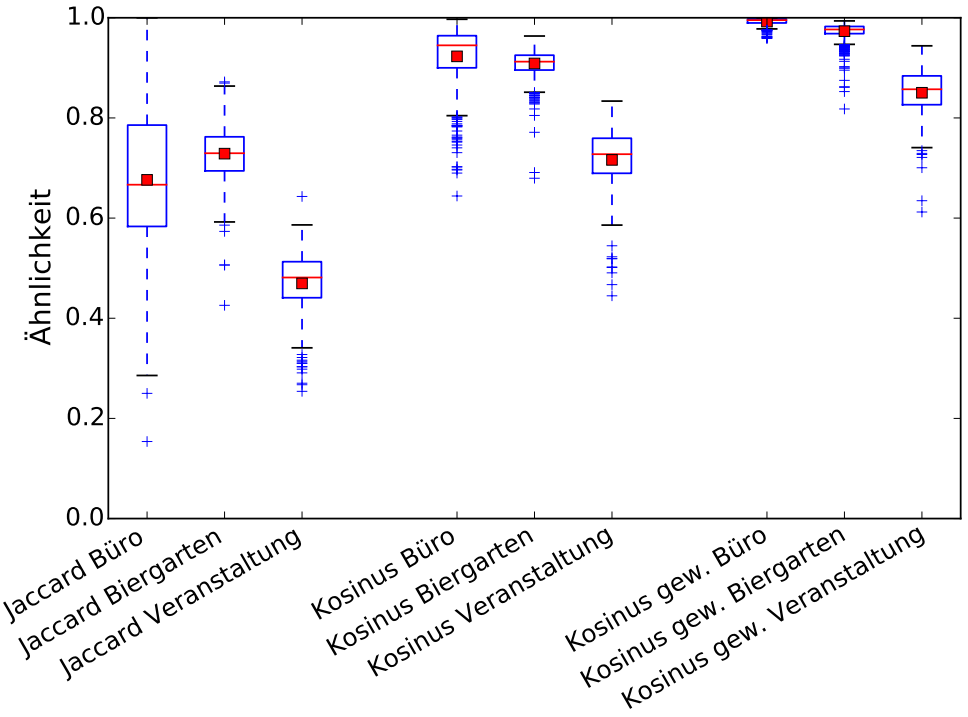


Abbildung 4.6: Verteilung der Ähnlichkeitswerte unterschiedlicher Ähnlichkeitsmaße in drei verschiedenen Szenarien.

weitere Eigenschaften zu untersuchen: Das Verhalten bei unterschiedlichen, größeren Distanzen und die eingangs geforderte Spezifizität bzgl. Ort und Zeit.

4.7.6 Distanzen

Um das Verhalten der Proximitätserkennung bei unterschiedlichen physischen Distanzen zu untersuchen, wurde ein weiteres Experiment durchgeführt. Auch in diesem Fall wurden zwei echte Testpersonen eingesetzt, die jeweils ein Testgerät in ihrer Hosentasche mit sich führten. Die Testpersonen mussten eine vordefinierte Route laufen, die teilweise durch einen öffentlichen Park und teilweise in einem Innenstadtbereich verlief. Dabei gab es fest definierte Vorgaben, in welchen Abschnitten die Personen zusammen gehen sollten, und in welchen eine Person stehen bleiben und die andere sich bis zu einem bestimmten Punkt entfernen sollte. Dadurch variierte der Abstand zwischen den Personen zwischen einem und 200 Metern. Während des Experiments wurden pro Zeitfenster im Schnitt 21 Signalquellen registriert, also etwas mehr als im Büro-Szenario, aber weniger als in den Biergarten- und Festival-Szenarien. Die Ergebnisse der Proxmitätsbestimmung sind in Abbildung 4.7 dargestellt.
Auf der Abszisse ist auch hier der zeitliche Verlauf des Versuchs angetragen, auf der linken Ordinate wiederum die errechneten Ähnlichkeitswerte. Zusätz-

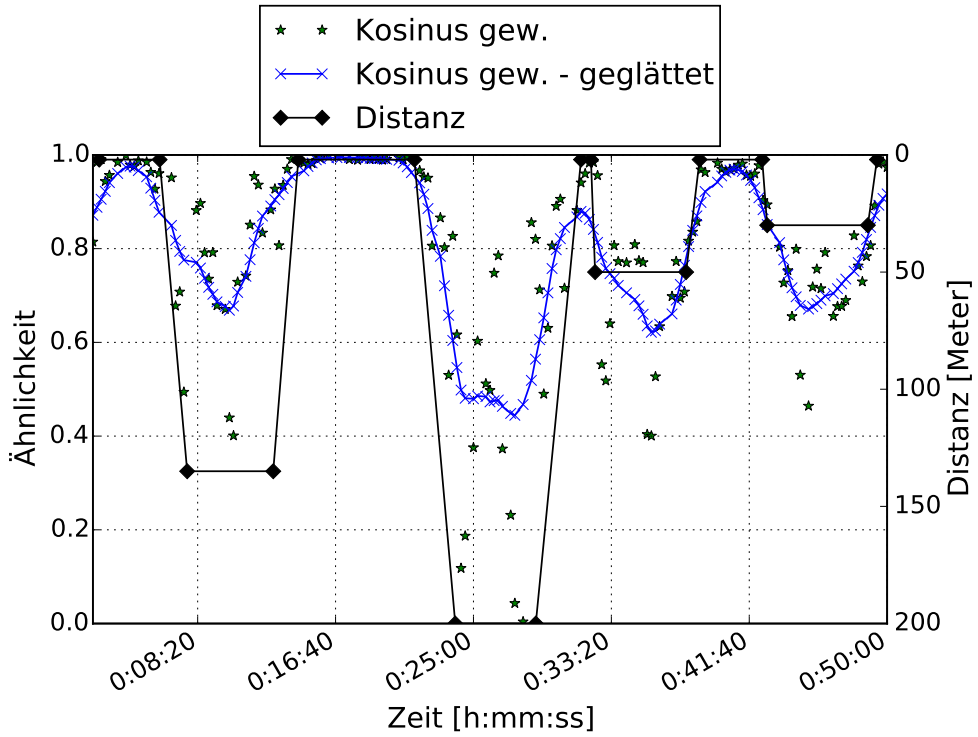


Abbildung 4.7: Ähnlichkeitswerte in unterschiedlichen räumlichen Entfernungen. Die geglättete Kurve folgt aus den Rohdaten durch Anwendung eines gleitenden Durchschnitts mit einer Fenstergröße von 7 und 2 Durchgängen.

lich existiert eine zweite Ordinate, die eine (umgekehrte) Skala für die tatsächliche physische Distanz der Testpersonen bietet. Im Plot finden sich als grüne einzelne Sterne die berechneten Ähnlichkeitswerte. Um den Verlauf etwas anschaulicher zu machen, gibt es zudem in blau eine geglättete Approximation der tatsächlichen Werte. Verwendet wurde hierzu ein gleitender Durchschnitt mit einer Fenstergröße von 7 Datenpunkten mit zwei Durchläufen. Parallel ist in schwarz der Verlauf der physischen Entfernung eingetragen.

Bei grober Betrachtung spiegeln sich die Änderungen der physischen Distanz in den Proximitätswerten wider. Entfernen sich die beiden Testpersonen voneinander, so sinken auch die Proximitätswerte zeitlich passend ab. Zeiträume, in denen sich die Personen zusammen fortbewegten, weisen sehr hohe Proximitätswerte nahe 1.0 auf. Im geglätteten Verlauf scheint dies beim lokalen Maximum bei ca. 30 Minuten nicht der Fall zu sein. Dies ist jedoch nur auf die Glättung und den zu kurzen gemeinsamen Zeitraum zurückzuführen.

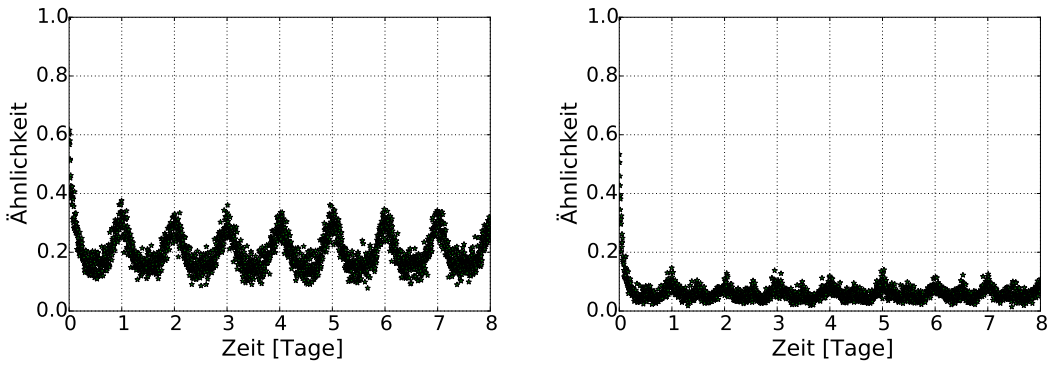
Aussagen über eine absolute Distanz zwischen den Testpersonen lassen sich anhand der Proximitätswerte nicht machen. Betrachtet man den geglätteten Verlauf, so ist zwar innerhalb eines Bewegungsablaufs, d.h. beim einmaligen voneinander Entfernen und wieder Zusammenkommen, auch der Verlauf der

Proximitätswerte passend und in den jeweiligen Abschnitten weitgehend monoton, konkrete Abbildung zwischen Proximitätswert und physischer Distanz sind jedoch nicht möglich. Dies zeigt sich vor allem beim ersten, dritten und vierten Distanz-Minimum: Trotz unterschiedlicher phyischer Distanzen verlaufen die Proximitätswerte relativ ähnlich.
Unter Berücksichtigung dieser Ergebnisse lässt sich schließen, dass das ProbeTag-Verfahren in unterschiedlichsten Szenarien sehr robust arbeitet, solange sich die Personen in direkter Nähe, d.h. innerhalb weniger Meter und mit Sichtkontakt befinden. Größere Entfernungen lassen sich nur schlecht einordnen, eine Abschätzung der Proximitätsentwicklung zwischen zwei Entitäten, d.h. aufeinander zu oder voneinander weg, ist in gewissem Umfang jedoch möglich. Der Grund für die nicht eindeutigen Zuordnungen von Proximitätswerten und physischen Distanzen bei größeren Entfernungen liegt vermutlich in der bei zunehmender Entfernung leichter möglichen Diversität der Situationen. Stehen zwei Personen direkt nebeneinander, so ist es sehr unwahrscheinlich, dass sich zwischen ihnen noch viele Hindernisse, z.B. andere Personen, befinden. Bei größerer Entfernung ist es wahrscheinlicher, dass die beiden individuellen Standorte durch Personen oder Objekte deutlicher voneinander getrennt sind bzw. gibt es hier eine größere Variabilität der Möglichkeiten. Auf freiem Gelände befinden sich bei 100 Metern Entfernung möglicherweise weniger Hindernisse zwischen den Entitäten als in einem Stadtgebiet bereits bei 20 Metern Entfernung. Diese Einflüsse kommen bei Entitäten, die sich sehr nahe beieinander befinden, seltener zustande.

4.7.7 Spezifität

Mit den bisher gezeigten Ergebnissen lässt sich folgern, dass das ProbeTag-Verfahren für bestimmte Anwendungsfälle eine zuverlässige Möglichkeit zur Proximitätsschätzung darstellt. Noch zu klären ist die Frage, inwieweit das System die essentiellen Anforderungen an Sicherheit und Privatsphäre erfüllt. Wie in den vorherigen Kapiteln gezeigt, sind Proximitätswerte oberhalb von 0.8 eine sehr zuverlässige Indikation von physischer Nähe zwischen den Parteien des Proximitätstests. Die entscheidende Frage ist, ob dies nur gilt, wenn sich beide Parteien zur gleichen Zeit am gleichen Ort befinden, oder ob ein vorher zu einem anderen Zeitpunkt aufgenommer ProbeTag zu einem späteren Zeitpunkt zur Vortäuschung von örtlicher Nähe erneut verwendet werden kann.
Um dies zu untersuchen, wurden zwei Langzeitmessungen untersucht. Der Datensatz wurde ursprünglich von Schauer et al. mit Ziel der Analyse von Besucherströmen aufgezeichnet [160]. Er enthält alle benötigten WLAN-Daten, sodass er sehr gut zur Evaluation des ProbeTag-Ansatzes geeignet ist. Für beide Messungen wurde je ein Testgerät jeweils 17 aufeinanderfolgende Tage lang stationär an einem öffentlichen Ort platziert. Die Testgeräte zeichneten durchgehend die WLAN-Signale der Umgebung auf.
Um die zeitliche Spezifität der ProbeTags zu evaluieren, wurde aus den auf-

gezeichneten Daten künstlich eine Testreihe erzeugt, die ProbeTags aus der gleichen Messung zu verschiedenen Zeitpunkten miteinander vergleicht, d.h. ein ProbeTag, der zum Zeitpunkt t aufgenommen wurde, wird mit dem ProbeTag aus dem gleichen Datensatz zum Zeitpunkt $t + \delta_t$ verglichen.
Es wurden beide Datensätze in zwei Hälften geteilt und dann aus der ersten Hälfte jeweils 50 ProbeTags zufällig gezogen. Für jeden dieser zufällig gezogenen ProbeTags wurde anschließend eine Testreihe generiert, in der der initiale ProbeTag mit allen späteren ProbeTags der Messung im Interval $[t, t + 8{,}5 \text{ Tage}]$ im Abstand von 300 Sekunden verglichen wurde. Somit entstanden pro Messreihe 50 zufällig ausgewählte Sub-Messreihen, die jeweils den Verlauf von ProbeTag-Ähnlichkeiten über 8,5 Tage enthalten. Diese 50 Sub-Messreihen wurden dann für jede Hauptmessreihe wieder zu einer gesamten Auswertung aggregiert, in dem die Ähnlichkeitswerte an den jeweils passenden Zeitpunkten gemittelt wurden. Die Ergebnisse der beiden Messungen sind in Abbildung 4.8 dargestellt.



(a) Standort 1 (b) Standort 2

Abbildung 4.8: Zeitliche Spezifität der ProbeTags; Ähnlichkeitswerte über die Zeit an zwei unterschiedlichen Standorten.

Auf der Abzisse ist in diesem Abbildungen erneut der zeitliche Verlauf angetragen, dieses Mal in Tagen, auf der Ordinate befinden sich die Ähnlichkeitswerte. In beiden Fällen ist zu erkennen, dass die Ähnlichkeiten bereits nach kurzer Zeit rapide abfallen und zu keiner Zeit mehr auch nur annähernd die initialen Ähnlichkeitswerte erreichen.
Wichtig sind zwei weitere Beobachtungen: Die Ähnlichkeitswerte sinken nie ganz auf 0 ab, und es existieren regelmäßige Maxima, die in Abständen von 24 Stunden auftreten. Ersteres ist auf stationäre WLAN-Endgeräte zurückzuführen, die an Verkaufsständen o.ä. installiert sein können und daher durchgehend an den jeweiligen Orten zu finden sind. Letzteres lässt sich durch analoge Regelmäßigkeiten im Alltag der Menschen erklären, d.h. beispielweise Personen, die täglich zur gleichen Zeit zur Arbeit gehen, u.ä.

Diese Testergebnisse ermöglichen zwei wichtige Erkenntnisse. Zum einen ist die erhoffte Schutzfunktion für die Privatsphäre der Benutzer nicht im geforderten Umfang gegeben. Zwar enthält ein ProbeTag keine absolute Ortsinformation an sich, es könnte jedoch mit Hilfe einer vorher aufgezeichneten Datenbank (ähnlich wie bei Ansätzen zur WLAN-Positionierung) ein Abgleich vorgenommen werden. Da sich wie gesehen die WLAN-Endgeräte an einem Ort zwar deutlich jedoch nicht komplett verändern, kann eine geringfügige Ähnlichkeit festgestellt werden, die eine Abschätzung des Aufenthaltsortes erlaubt.
Zum anderen zeigen die Ergebnisse auch klar, dass die durch vorher aufgezeichnete ProbeTags fälschbaren Ähnlichkeitswerte deutlich unter den Werten liegen, die zwei Endgeräte produzieren, die sich tatsächlich zur gleichen Zeit am gleichen Ort befinden. Es ist also bei Verwendung des ProbeTag-Verfahrens sehr schwierig bis unmöglich, der anderen Partei eines Proximitätstests eine nicht vorhandene Proximität vorzutäuschen. Damit erfüllt das Proximitätserkennungsverfahren für sich allein stehend die Anforderung bzgl. Privatsphäre eingeschränkt, die noch wichtigere Anforderung bzgl. Fälschungssicherheit jedoch vollumfänglich.

4.8 Diskussion und Zusammenfassung

In diesem Kapitel wurde ein erstes Verfahren zur Erkennung räumlicher Proximität vorgestellt. Das Verfahren basiert auf der Erstellung von Location Tags mit Hilfe von WLAN-Signalen, die von Endgeräten in der Umgebung der Benutzer ausgesandt werden. In der Evaluation konnte gezeigt werden, dass das Verfahren eine grundsätzliche Abschätzung räumlicher Nähe ermöglicht, insbesondere in einem sehr nahen Bereich von ca. fünf Metern. Es benötigt keine dedizierte Infrastruktur und kann auf aktueller Standardhardware umgesetzt werden. Das Verfahren zielt allein auf die Erkennung von räumlicher Nähe ab, die Erkennung weitergehender Kontextähnlichkeiten (wie beispielweise ähnliche Aktivitäten) ist nicht explizit Teil des Ansatzes.
Im Hinblick auf die primären Anforderungen bzgl. Unfälschbarkeit der Location Tags sowie der Wahrung der Privatsphäre der Benutzer weist das Verfahren im Vergleich zu verwandten Arbeiten sehr gute Eigenschaften auf. Durch ein Langzeit-Experiment konnte gezeigt werden, dass es – zumindest in einem öffentlichen Raum – sehr unwahrscheinlich ist, dass wiedereingespielte, in der Vergangenheit aufgezeichnete ProbeTags, einen ausreichend hohen Ähnlichkeitswert erzeugen, um eine nicht vorhandene Proximität vortäuschen zu können. Die ProbeTags enthalten zudem keine absoluten Ortsinformationen. Sie sind jedoch evtl. anfällig für die Rekonstruktion des tatsächlichen Ortes mit Hilfe von WLAN-Fingerprint-Datenbanken.
Die letztgenannte kleine Schwäche des Verfahrens ließe sich beheben, indem das ProbeTag Verfahren mit einem der in Kapitel 3.2.6 vorgestellten Verfahren zum PPT kombiniert wird. Mit Hilfe des PPT-Verfahrens könnte sichergestellt werden, dass die eigenen Eingabedaten eines Benutzers in den Pro-

ximitätstest nicht offengelegt werden. Dadurch kann ein Vergleich der Eingabendaten mit einer Fingerprint-Datenbank verhindert werden. Durch das ProbeTag-Verfahren wäre in dieser Kombination wiederum sichergestellt, dass keine Orte „geraten“ werden können, um durch diese gefälschten Ortsangaben die Sicherheit und Privatsphäre der anderen Nutzer zu untergraben.
Darüberhinaus gehen die Hersteller mobiler Endgeräte dazu über, die MAC-Adressen, die die Netzwerkschnittstellen beim Versenden u.a. von Probe Requests verwenden, zu randomisieren [91]. Das Ziel dabei ist, das Tracking der Benutzer zu verhindern. Würde diese Technik in der Zukunft weitflächig eingesetzt, so würde sich das überaus positiv auf die Eigenschaften des ProbeTag-Verfahrens auswirken. In der jetzigen Form ist die Spezifität der ProbeTags, d.h. im Prinzip die Zufälligkeit dieser, hauptsächlich von der Mobilität der Menschen abhängig, d.h. dass sich nie die selben Menschen in der selben Konstellation am selben Ort befinden. Würden jetzt jedoch zusätzlich die MAC-Adressen zufällig gewechselt, ergäbe dies eine enorm wertvolle weitere Entropiequelle für die ProbeTags, die das Verfahren insbesondere im Hinblick auf Fälschungssicherheit und Privatsphäre nochmals robuster machen würde.

5 Proximitätserkennung mit visuellen Featurepunkten

Nachdem im vorherigen Kapitel ein erster Ansatz zur Bestimmung räumlicher Proximität vorgestellt wurde, folgt in diesem Kapitel ein weiterer Ansatz, der jedoch nicht nur die räumliche Nähe der Benutzer sondern auch deren Übereinstimmung der Aktivität implizit berücksichtig. Die Grundidee besteht darin, dass Personen, die zusammen unterwegs sind – d.h. eine Gruppe bilden – in vielen Fällen auch die gleichen Objekte in der Umgebung sehen. Könnte man also das Blickfeld einer Person, in dem sich u.a. andere Personen, Gegenstände und Gebäude befinden, analysieren und mit den Blickfeldern anderer Testpersonen vergleichen, so ließe sich dadurch eine kontextuelle Proximität ableiten. Durch das vermehrte Aufkommen von Wearables wie Action-Kameras oder Smart Glasses erscheint eine entsprechende Möglichkeit für die Zukunft mehr als wahrscheinlich.

Das im Folgenden vorgestellte Verfahren basiert darauf, die mit einer am Körper getragenen Kamera aufgezeichneten Bilddaten verschiedener Personen in eine abstrakte, Orts-Semantik-lose Repräsentation zu überführen und diese zum Vergleich, d.h. Proximitätstest, mit weiteren Teilnehmern zu verwenden.

Das Kapitel gliedert sich wie folgt: In Abschnitt 5.2 wird zunächst die Grundidee des Verfahrens ausführlicher erläutert, bevor in Abschnitt 5.3 wichtige Grundlagen zum Verständnis des Verfahrens erklärt werden. Danach folgt ein kurzer Überblick über weitere verwandte Arbeiten (Abschnitt 5.4). In Abschnitt 5.5 wird das Konzept des neuen Ansatzes vorgestellt und dieses dann in Abschnitt 5.6 ausführlich evaluiert. Abschließend folgt in Abschnitt 5.7 eine Diskussion und Zusammenfassung.

5.1 Vorveröffentlichungen

Die Kerninhalte dieses Kapitels wurden vom Autor bereits in [126] publiziert und im Rahmen der vorliegenden Arbeit erweitert und überarbeitet. Wie in Kapitel 1.3 ausführlich dargestellt, stammen die im Paper und im Folgenden präsentierten Inhalte bzgl. der Idee, des Konzepts und der Evaluation vom Autor der vorliegenden Arbeit. Die Inhalte bzgl. der Logik-Ebene (vgl. Abschnitt 5.5.6) sowie die Evaluation mit Hilfe der Simulationsumgebung (vgl. Abschnitt 5.6.1) waren im Paper noch nicht enthalten.

5.2 Motivation und Grundidee

Die grundsätzliche Idee des folgenden Ansatzes zur Proximitätserkennung liegt in der Verwendung visueller Informationen, die durch Benutzung einer am Kopf bzw. generell am Körper getragenen Kamera gewonnen und analysiert werden können. Das Blickfeld eines teilnehmenden Nutzers wird ausgewertet, um daraus eine Beschreibung der Umgebung abzuleiten, die mit anderen Nutzern zur Durchführung eines Vergleichs ausgetauscht werden kann.
Die Grundannahme für das Verfahren besteht darin, dass Personen, die sich in einem engen gemeinsamen Kontext befinden, auch die selben Dinge sehen. Die abstrakte Defintion von Proximität geht hier also einen Schritt weiter als beim in Kapitel 4 vorgestellten ProbeTag-Verfahren. Reine örtliche Nähe ist in diesem Fall nicht ausreichend, vielmehr müssen auch die Aktivitäten der Personen insoweit übereinstimmen, dass sie die gleichen Objekte betrachten. Letzteres ist insbesondere der Fall, wenn die Testpersonen sich gemeinsam und zu Fuß fortbewegen. Gerade in öffentlichen Räumen in größeren Menschenmengen bewegen sich bis zu 70% der Menschen zusammen in kleinen Gruppen [138]. In diesem Szenario ist sowohl eine große örtliche Nähe gegeben als auch eine starke Ähnlichkeit der Aktivität, es ist daher für die weiteren Ausführungen der primär betrachtete Anwendungsfall. Es werden im weiteren Verlauf jedoch auch noch anderen Szenarien betrachtet.
Da es weder aus Privatsphäre- noch aus Bandbreiten-Sicht sinnvoll wäre, komplette Bilder bzw. Videos auszutauschen, müssen die visuellen Informationen in eine geeignetere Repräsentation überführt werden. Zur Repräsentation der visuellen Informationen wird ein Feature-Punkt-Verfahren angewendet. Feature-Punkt-Verfahren versuchen in einem Bild interessante, d.h. charakteristische, Punkte zu identifizieren und extrahieren eine kompaktere Beschreibung des Bildes anhand dieser Punkte. So beschriebene Bilder können auf Basis der Feature-Punkt-Repräsentation verglichen werden, um ähnliche oder identische Bilder zu finden. Da für das hier vorgestellte Verfahren der SURF-Algorithmus zur Extraktion von Feature-Punkten eingesetzt wird, trägt das System den Namen *SURFtogether*. Analog zu den in Kapitel 4 vorgestellten ProbeTags können auch die im SURFtogether-Verfahren eingesetzten Merkmalsvektoren – bestehend aus visuellen Feature-Punkten, welche die Umgebung beschreiben, sowie weiteren Sensor-Informationen – als Location Tag aufgefasst werden.

In Abbildung 5.1 sind ein grundsätzliches Szenario und Ablauf zur Berechnung der Proximität visualisiert. Die beiden Benutzer Alice und Bob sind zusammen unterwegs und von weiteren Personen und unterschiedlichen Objekten umgeben. Es soll nun ein Proximitätstest zwischen Alice und Bob durchgeführt werden, inbesondere möchte Bob erfahren, ob Alice sich in der Nähe befindet. Beide Benutzer verfügen über ein am Körper getragenes Endgerät mit integrierter Kamera und Sensorik, das es erlaubt, die jeweiligen Blickfelder zu analysieren. Die Blickfelder der beiden Perso-

nen überschneiden sich in gewissem Maße, sodass sie teilweise die selben Personen und Objekte sehen.

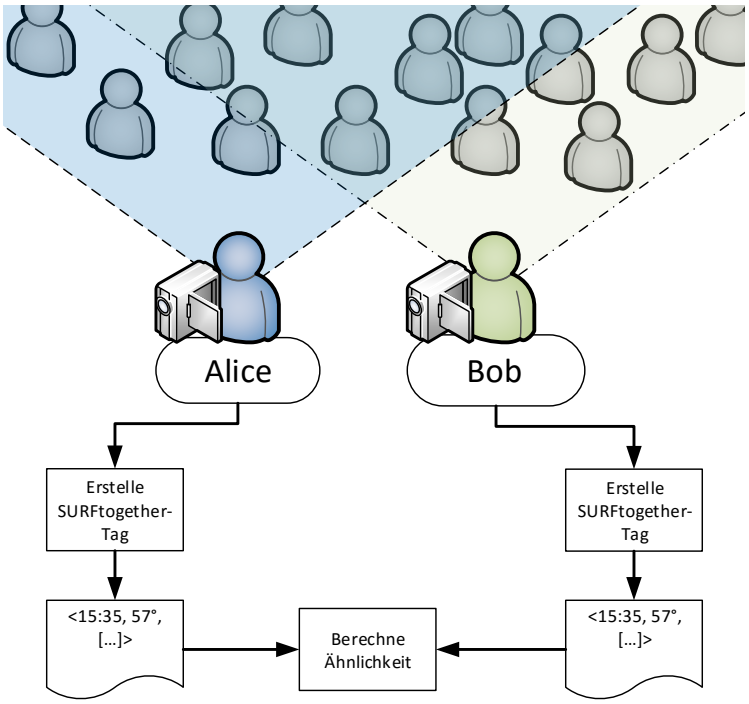


Abbildung 5.1: Beispielszenario und Ablauf des SURFtogether-Verfahrens. Alice und Bob möchten die Proximität bestimmen. Beide können durch eine am Körper getragene Kamera das aktuelle Blickfeld aufzeichnen und auswerten. In der Umgebung befinden sich verschiedene Personen, die teilweise von beiden gleichzeitig gesehen werden.

5.3 Grundlagen zu Merkmalspunkt-Verfahren

Der vorgestellte Ansatz zur Proximitätserkennung benutzt zur abstrakten Repräsentation und zum Vergleich der Bilder ein sogenanntes Featurepunkt- bzw. Merkmalspunkt-Verfahren. Im Folgenden wird ein kurzer Überblick über diese Verfahren allgemein, sowie über das verwendete SURF-Verfahren im Speziellen gegeben.

5.3.1 Merkmalspunkt-Verfahren

Die Berechnung verschiedener Merkmale von Bildern wird allgemein im Bereich der Bildverarbeitung und Computer Vision eingesetzt. Typische Merkmale sind dabei Kanten oder Ecken in Bildern. Für den vorliegenden Fall sind besonders solche Verfahren interessant, die sogenannte *Merkmalspunkte* (engl.: *Feature*

Points oder *Interest Points (IPts)*) verwenden. Für den praktischen Einsatz sind i.d.R. drei Schritte notwendig:

- Erkennung der Merkmalspunkte (*Detektor*)
- Beschreibung der Merkmalspunkte (*Deskriptor*)
- Vergleich der Merkmalspunkte

Es existiert keine allgemeingültige Definition dafür, wann eine Stelle eines Bildes „interessant“ ist. Mögliche Definitionen beziehen sich z.B. auf die Entropie in der Umgebung eines Bildpunktes [162]. Im Allgemeinen gilt jedoch, dass ein Verfahren sinnvolle Merkmalspunkte extrahiert, wenn der darauf aufbauende Anwendungsfall gut funktioniert [166].
Sind die interessanten Punkte eines Bildes identifiziert, so müssen diese auf geeignete Art und Weise beschrieben werden. Der einfachste Fall besteht darin, einen lokalen Bildausschnitt um den entsprechenden Punkt zu verwenden. Diese Variante hat jedoch den Nachteil, dass die Beschreibung weder skalierungs- noch rotationsinvariant ist. Da für den vorgesehenen Anwendungsfall von den Benutzern zwar die gleiche Szene beobachtet wird, dies jedoch aus unterschiedlichen Positionen und Winkeln geschieht, muss eine abweichende Skalierung und/oder Rotation kompensiert werden können. Geeignetere Verfahren berücksichtigen daher diese beiden Transformationen, um eine robustere Bildbeschreibung abzuleiten [61]
Die durch eine Menge von Merkmalspunkten beschriebenen Bilder können nun verglichen werden. Welche Vergleichsfunktion hierbei zum Einsatz kommt, hängt von der Art der Beschreibung der Bilder ab.
Beispiele für solche Merkmalspunkt-Verfahren sind GLOH [135], BRIEF [32], BRISK [115] und SIFT [120, 121]. Letzteres bildet die Basis für das in dieser Arbeit verwendete SURF-Verfahren.

5.3.2 Speeded Up Robust Features (SURF)

Es ist i.d.R. vom Anwendungsfall abhängig, welches der vielen verschiedenen Merkmalspunkt-Verfahren am geeignetsten für das gewählte Szenario ist. Ein skalierungs- und rotationsinvariantes Verfahren, das durch die gute Performanz besonders für den Einsatz auf mobilen Endgeräten geeignet ist [61], ist das *Speeded Up Robust Features (SURF)*-Verfahren [19, 18].
Das SURF-Verfahren basiert grundsätzlich auf dem älteren SIFT-Verfahren [120, 121] und ersetzt einige Teile von diesem, um die Performance zu verbessern. Während das SIFT-Verfahren zur Detektion interessanter Punkte einen Gauß-Filter verwendet, benutzt SURF eine Annäherung dessen in Form eines Mittelwertfilters (engl.: *Box Filter*). Durch die Verwendung von Integralbildern ist die Berechnung der Punkte in nahezu konstanter Zeit unabhängig von der Größe des Bildes möglich.

Der verwendete Deskriptor besteht aus einem 64-stelligen Vektor, der die Umgebung, Position, Skalierung und Rotation des entsprechenden Punktes beschreibt. Der Vergleich der Punkte erfolgt durch Berechnung der euklidischen Distanz zwischen ihren Deskriptoren. Ein „Match“, d.h. die Gleichheit von Punkten, ist dabei anhand eines Nearest-Neighbor-basierten Verfahrens definiert [162]. Für die Punkte A, B und C bzw. ihre Deskriptoren D_A, D_B und D_C gilt, dass A und B ein Match sind, falls D_B der nächste Nachbar von D_A und D_C der zweitnächste Nachbar von D_A ist und es gilt:

$$\frac{|D_A - D_B|}{|D_A - D_C|} < t \tag{5.1}$$

D.h. A und B werden als identisch betrachtet, wenn sie die zueinander ähnlichsten Punkte der Gesamtmenge sind und dabei der zweitähnlichste Punkte in gewissem Maße unähnlicher ist.

5.4 Verwandte Arbeiten aus dem Bereich der visuellen Positionierung

Neben den bereits in Kapitel 3 beschriebenen Ansätzen zur Proximitätserkennung, u.a. basierend auf Location Tags, ist speziell für das hier vorgestellte Verfahren ein Klasse von Ansätzen zusätzlich relevant: Verfahren zur visuellen (Indoor-)Positionierung.

Die Grundidee besteht dabei darin, eine Datenbank von Bildern zusammen mit Meta-Daten wie insbesondere dem Aufnahmeort zu erstellen, und diese dann zur Positionsbestimmung zu verwenden. Dazu nimmt ein Benutzer bei Bedarf ein Bild auf und vergleicht dieses mit den aufgezeichneten Daten. Wird eine Übereinstimmung gefunden, so kann der aktuelle Ort des Benutzers abgeleitet werden.

Die Datenbank kann dabei entweder durch ein *Simultaneous Localization and Mapping (SLAM)*-Verfahren im laufenden Betrieb dynamisch [97] oder ähnlich wie bei vielen WLAN-Positionierungsansätzen in einem vorhergehenden Schritt offline erstellt werden [163, 194].

Die Verfahren stützen sich im Allgemeinen auf Aufnahmen von statischen, dominanten Landmarken wie große Gebäude. Bewegliche Objekte wie Menschen führen dagegen zu einer deutlichen Verschlechterung der Positionierungsmöglichkeiten, da diese nicht in den aufgezeichneten Daten vorhanden sind und damit die Bildinformationen „verfälschen“. Ähnlich wie bei dem in Kapitel 4 vorgestellen Ansatz sind es aber genau diese Störungen, die für den Einsatz als Proximitätserkennungsverfahren vielversprechend sind.

5.5 Konzept zur Proximitätserkennung mit visuellen Featurepunkten

Wie bereits beschrieben, basiert das SURFtogether-Verfahren auf der Idee, Merkmale aus dem Videomaterial einer am Körper getragenen und ungefähr das Blickfeld des Benutzers aufnehmenden Kamera zu extrahieren und als Grundlage für einen Proximitätstest zu verwenden. Analog zum ProbeTag-Verfahren (Kapitel 4) besteht der grundsätzliche Ablauf aus zwei Phasen: i) Der lokalen Aufzeichnung und Erstellung der Umgebungsrepräsentation und ii) dem Vergleich der Merkmalsvektoren verschiedener Nutzer.

5.5.1 Konstruktion von SURFtogether-Tags

Die im SURFtogether-Verfahren für die Proximitätsbestimmung generierten Informationspakete können analog zu den ProbeTags als Location Tags aufgefasst werden. Daher werden sie im Folgenden zur einfacheren Referenzierung als SURFtogether-Tags (ST-Tags) bezeichnet. Ein ST-Tag beschreibt die Umgebung des Benutzers zu einem bestimmten Zeitpunkt. Dabei spielt auch die Aktivität des Benutzers eine Rolle, da diese den betrachteten und damit auswertbaren Ausschnitt der Umgebung (d.h. die Blickrichtung) beeinflusst.

Ein ST-Tag besteht aus drei Informationseinheiten: Einem Zeitstempel, der Blickrichtung und dem SURF-Merkmalsvektor. Der Zeitstempel und die Blickrichtung werden dabei von der internen Uhr und dem internen Kompass bzw. Magnetometer des Endgeräts geliefert, der SURF-Merkmalsvektor wird aus den aktuellen Bilddaten der Kamera des Endgeräts errechnet. Die Extraktion von SURF-Punkten wurde bereits in Kapitel 5.3.2 beschrieben.

Soll ein Proximitätstest durchgeführt werden, so müssen beide Parteien des Tests einen aktuellen ST-Tag erstellen. Dazu werden die Daten von Uhr und Kompass ausgelesen und ein Bild mit Hilfe der Kamera aufgenommen. Auf das Bild wird das SURF-Verfahren angewandt und der resultierende Merkmalsvektor zusammen mit Zeitstempel und Blickrichtung zum ST-Tag zusammengefasst.

Auf der passiven Seite eines Proximitätstests, d.h. für den Fall, dass nicht man selbst sondern eine andere Entität die Proximität überprüfen will, reicht es, bei Bedarf den aktuellen ST-Tag zu generieren und zu übermitteln. Eine längere Datenaufzeichnung wie beim ProbeTag-Verfahren ist zunächst nicht notwendig. Auf der aktiven Seite eines Proximitätstests kann es notwendig sein, auf frühere ST-Tags zugreifen zu können. Die Gründe dafür werden im nächsten Kapitel im Rahmen der verschiedenen Vergleichsverfahren beschrieben. In diesem Fall ist also eine kontinuierliche Generierung und teilweise Speicherung der lokalen ST-Tags notwendig. Es kann auch auf der passiven Seite aus Performanzgründen sinnvoll sein, ST-Tags kontinuierlich zu erzeugen und vorzuhalten. Dies wird im Folgenden jedoch nicht weiter betrachtet.

5.5.2 Vergleich von SURFtogether-Tags

Die Erstellung der ST-Tags verläuft wie gezeigt sehr geradlinig. Die eigentliche Intelligenz des Verfahrens kommt beim Vergleich der ST-Tags zum Tragen. Für diesen Vergleich wird im Folgenden zunächst ein Basisverfahren vorgestellt, das dann mit verschiedenen Erweiterungen robuster und performanter gemacht wird.

5.5.3 Basisverfahren zur Proximitätserkennung

Das Basisverfahren ist für den Standardfall ausgelegt, dass zwei Benutzer zusammen sind und in die gleiche Richtung blicken. Dadurch haben sie zum gleichen Zeitpunkt die gleichen Objekte im Blickfeld und nehmen mit der Kamera ein sehr ähnliches Bild auf. Bei diesem grundsätzlichen Vorgehen wird die im ST-Tag enthaltene Blickrichtung noch nicht berücksichtigt. Der Vergleich basiert allein auf den beiden SURF-Merkmalsvektoren, die zum gleichen Zeitpunkt aus dem jeweiligen Kamerabild extrahiert wurden.
Dieses grundsätzliche Verfahren gliedert sich in vier Teilschritte:

- Überprüfung der Anzahl der Merkmalspunkte
- Berechnung der Übereinstimmungen
- Berechnung des Modifikators
- Berechnung der Ähnlichkeit

Die in den ST-Tags enthaltenen Merkmalsvektoren zweier Bilder Bild_A und Bild_B werden im Folgenden als Ipts_A ($\text{Ipts}_A = \text{SURF}(\text{Bild}_A)$) und Ipts_B ($\text{Ipts}_B = \text{SURF}(\text{Bild}_B)$) bezeichnet.

Überprüfung der Anzahl der Merkmalspunkte Im ersten Schritt wird überprüft, ob ein sinnvoller Vergleich überhaupt möglich ist. Dies hängt von der Anzahl der gefundenen Merkmalspunkte ab, die für beide Bilder über einem bestimmten Minimum th_{min} liegen muss, d.h. es muss die folgende Bedingung erfüllt sein:

$$|\,\text{Ipts}_A\,| > \text{th}_{\text{min}} \wedge |\,\text{Ipts}_B\,| > \text{th}_{\text{min}} \tag{5.2}$$

Wird dieser Grenzwert zu niedrig angesetzt, kann dies die Anzahl der falsch-positiven Resultate erhöhen, die durch zufällige Ähnlichkeiten bei nur sehr wenigen gefundenen Merkmalspunkten auftreten können. Wird er dagegen zu hoch gewählt, werden valide hohe Ähnlichkeiten verworfen und damit die falsch-negativen Resultate erhöht.
Eine minimale Anzahl an Merkmalspunkten wird vorausgesetzt, um damit Bilder, die wahrscheinlich „beschädigt“ sind, von vornherein auszuschließen. Ein

typisches Beispiel dafür sind Bilder, die durch einen Softwarefehler stark überbelichtet wurden und damit fast nur eine weiße Fläche zeigen. Dies kann z.B. beim Wechsel von Innen- zu Außenbereichen kurzzeitig auftreten. Solche Bilder sind für einen Vergleich nicht geeignet und können damit sofort ausgeschlossen werden. In vorausgehenden Experimenten hat sich gezeigt, dass $\mathrm{th}_{\min} = 5$ eine sinnvolle Belegung ist, um viele solcher nicht verwendbarer Bilder zu identifizieren. Gleichzeitig ist diese Grenze niedrig genug, dass jedes valide Bild darüber liegt.
Sollte die Mindestanzahl an Merkmalspunkten in einem Fall nicht erreicht sein, so wird 0%-Ähnlichkeit als Ergebnis zurückgeliefert. Ist die Bedingung erfüllt, so wird der nächste Schritt ausgeführt.

Berechnung der Übereinstimmungen Aus den beiden Merkmalsvektoren werden nun die gemeinsamen Merkmalspunkte beider Vektoren ($\mathrm{Ipts}_\cap$) extrahiert:

$$\mathrm{Ipts}_\cap = \mathrm{Ipts}_A \cap \mathrm{Ipts}_B \tag{5.3}$$

Wie in Kapitel 5.3.2 beschrieben, wird die Gleichheit von Merkmalspunkten anhand des Nächste-Nachbarn-basierten Verfahrens Bay et al. [18, 162] bestimmt.
Für beide Merkmalsvektoren wird dann das Übereinstimmungsverhältnis Match_A bzw. Match_B berechnet, d.h.

$$\mathrm{Match}_A = \frac{|\,\mathrm{Ipts}_\cap\,|}{|\,\mathrm{Ipts}_A\,|} \tag{5.4}$$

$$\mathrm{Match}_B = \frac{|\,\mathrm{Ipts}_\cap\,|}{|\,\mathrm{Ipts}_B\,|} \tag{5.5}$$

Damit existiert eine erste Einschätzung der Ähnlichkeit der beiden Merkmalsvektoren.

Berechnung des Modifikators Im Allgemeinen kann beobachtet werden, dass die Anzahl der gefundenen Merkmalspunkte beim SURF-Verfahren stark vom analysierten Bildmaterial abhängt. Für den für das SURFtogether-Verfahren relevanten Anwendungsfall, d.h. Personen, die sich in der Welt bewegen, zeigt sich dieser Unterschied vor allem im Vergleich von Innen- und Außenbereichen. Es ist intuitiv nachvollziehbar, dass Innenbereiche, die in alle Richtungen eine beschränktere Sichtweite aufweisen – die zudem meist von sehr monotonen, d.h. einfarbigen und weitgehend strukturlosen, Wänden begrenzt ist – weniger signifikante Punkte hevorbringen als weitläufige Außenbereiche. Dies ist aber für den betrachteten Anwendungsfall genau die Spannweite der Situationen, die auftreten können.
Das Problem hierbei ist, dass eine sehr hohe Anzahl an gefundenen Merkmalspunkten die berechneten Übereinstimmungsverhältnisse sehr stark negativ be-

einflusst. Werden in einem Innenbereich beispielsweise 5 übereinstimmende Punkte bei einer Gesamtanzahl von 50 Merkmalspunkten gefunden und im Vergleich dazu in einem Außenbereich 50 Übereinstimmungen bei 500 Merkmalspunkten insgesamt, so ergibt sich das gleiche Übereinstimmungsverhältnis. 50 übereinstimmende Punkte sind jedoch absolut betrachtet ein sehr hoher Wert, der sich im finalen Ergebnis positiver auswirken soll als nur 5 gefundene gemeinsame Punkte.
Um der absoluten Anzahl an gemeinsamen Merkmalspunkten mehr Einfluss auf das Ergebnis der Ähnlichkeitsberechnung zu verleihen, wird ein Modifikator m eingeführt. Vorausgehende Experimente haben gezeigt, dass m maximal eine Halbierung bzw. Verdopplung des eigentlichen Ergebnisses bewirken sollte, d.h. $m \in [0.5, 2.0]$. m berechnet sich dabei in Abhängigkeit eines Grenzwertes für eine „neutrale“ Anzahl an gemeinsamen Merkmalspunkten th_m wie folgt:

$$m = \max(0.5, \min(2.0, \frac{|\,\mathrm{Ipts}_\cap\,|}{\mathrm{th}_m})) \tag{5.6}$$

Es wird also das Verhältnis von gefundenen gemeinsamen Merkmalspunkten zu einer neutralen Grenze berechnet, um damit später den berechneten Ähnlichkeitswert zu verstärken oder abzuschwächen. Wie bereits erwähnt, wird m nach oben und unten begrenzt.
Werden deutlich weniger gemeinsame Merkmalspunkte als th_m beim Vergleich ähnlicher Bilder gefunden, so ist dies für das betrachtete Szenario eher ungewöhnlich und soll daher zu einer Abwertung führen. Umgekehrt ist eine Aufwertung sinnvoll, wenn deutlich mehr Punkte gefunden werden.

Berechnung der Ähnlichkeit Mit den in den vorherigen Schritten berechneten Werten wird nun im letzten Schritt der Ähnlichkeitswert der beiden Merkmalsvektoren $\mathrm{sim}(\mathrm{Ipts}_A, \mathrm{Ipts}_B)$ bestimmt. Dies geschieht nach der Formel

$$\mathrm{sim}(\mathrm{Ipts}_A, \mathrm{Ipts}_B) = \min(1.0, m\frac{\mathrm{Match}_A + \mathrm{Match}_B}{2}) \tag{5.7}$$

Der so berechnete Ähnlichkeitswert der Merkmalsvektoren wird als Ergebnis der Ähnlichkeitsberechnung der beiden ST-Tags zurückgeliefert.

5.5.4 Warteschlangen-Erweiterung

Beim Basisverfahren wird bisher vom Idealfall ausgegangen, dass die Uhren der Endgeräte der Teilnehmer des Proximitätstests exakt synchronisiert sind und dass es auch immer am besten ist, genau zeitgleich aufgenomme Bilder zu vergleichen. Beide Annahmen sind nicht immer zutreffend.
Es ist für das SURFtogether-Verfahren nicht notwendig, das Bildmaterial in der Geschwindigkeit von Videoaufnahmen, d.h. mit 24 Bildern pro Sekunde oder schneller, aufzunehmen und auszuwerten. Jedoch sollten zumindest

Bildraten von einem bis zehn Bildern pro Sekunde erreicht werden. Folglich müssten die Uhren sekundengenau oder noch präziser synchronisiert sein, was in verteilten Systemen eine Herausforderung darstellt.
Auch die Annahme, dass genau zeitgleich gemachte Aufnahmen die beste Grundlage für einen Vergleich sind, ist in der Praxis oft falsch. Kurzzeitige Abweichungen des Blickwinkels, „ins Bild laufende“ Personen und andere sehr kurzfristige Einflüsse auf das Blickfeld treten in einer realen Umgebung sehr häufig auf. Es kann also in vielen Fällen sein, dass die sich zusammen befindenden Teilnehmer eines Proximititätstests sehr ähnliche Aufnahmen machen, jedoch – bedingt durch Synchronisationsungenauigkeiten oder äußere Einflüsse – zu leicht unterschiedlichen Zeitpunkten.
Um diese Effekte im SURFtogether-Verfahren zu berücksichtigen, kann es sinnvoll sein, das Basisverfahren um ein Warteschlangensystem zu erweitern. Hierzu hält jeder Teilnehmer eine Warteschlange einer bestimmten Länge mit den in der Vergangenheit erzeugten ST-Tags vor. Die Warteschlange ist nach dem First-in-First-out (FIFO)-Prinzip organisiert, sodass bei Erreichen der maximalen Länge ein neues Bild automatisch das älteste verdrängt. Teilnehmer, die nicht aktiv einen Proximitätstest durchführen, sondern die örtliche Nähe nur einem anderen Teilnehmer beweisen wollen, können auf das Anlegen der Warteschlange verzichten. In diesem Fall reicht die Übermittlung des aktuellen ST-Tags.
Bei Benutzung der Warteschlangenerweiterung wird also ein ST-Tag eines anderen Nutzers nicht nur mit dem eigenen aktuellen ST-Tag verglichen, sondern mit allen ST-Tags aus der Warteschlange. Jeder dieser Vergleiche läuft wiederum nach dem Basisverfahren ab, und es wird final der Maximalwert aller so berechneten Ähnlichkeiten als Ähnlichkeitswert zurückgegeben.

5.5.5 Sektorbezogene Warteschlagen-Erweiterung

Beim Basisverfahren wird bisher die Orientierung, unter der die zugehörigen Bilder der zu vergleichenden Merkmalsvektoren aufgenommen wurden, nicht berücksichtigt. Es ist selbst im primären betrachteten Anwendungsfall, dass sich die Personen zusammen zu Fuß fortbewegen, sehr wahrscheinlich, dass sie sehr oft in unterschiedliche Richtungen blicken. In anderen Szenarien, z.B. wenn die beiden Personen gemeinsam an einem Tisch sitzen, kann es für längere Zeiträume vorkommen, dass die Personen nicht die gleichen Objekte im Blickfeld haben, obwohl sie sich im gleichen Kontext sehr nah zusammen befinden. Trotzdem würden sie von Zeit zu Zeit, z.B. beim Betreten des Raumes oder kurz vor dem Platz nehmen am Tisch, sehr ähnliche Blickwinkel besitzen. Mit dem Basisverfahren allein würden Proximitätstests in solchen Situationen nur sehr selten positiv ausfallen und damit zu vielen falsch-negativen Ergebnissen führen. Die Warteschlangen-Erweiterung berücksichtigt implizit unterschiedliche Blickrichtungen, sofern deren Auftreten nicht weiter in der Vergangenheit liegt als durch die Länge der Warteschlange abgebildet. Sie ist jedoch

zum Ausgleich von Synchronisationsungenauigkeiten und ungünstigen Aufnahmezeitpunkten gedacht und beeinträchtigt mit zunehmender Länge durch die immer größer werdende Anzahl von ST-Tag-Vergleichen die Performanz. Warteschlangen sollten im Allgemeinen also eher kurz gewählt werden und bieten damit nicht den nötigen Umfang, um die Problematik durch unterschiedliche Orientierungen auszugleichen.
Um die Blickrichtungsunterschiede besser zu berücksichtigen und auszugleichen, wird im Folgenden die sogenannte Sektorbezogene Warteschlangen-Erweiterung (SWE) vorgestellt. Die Grundidee dieser Erweiterung besteht darin, nicht nur eine Warteschlange vorzuhalten, sondern mehrere, verteilt auf unterschiedliche Orientierungssektoren. Neu erzeugte ST-Tags werden also anhand der Orientierung einem Sektor zugeordnet und in die entsprechende Warteschlange eingefügt. Soll nun ein Vergleich mit einem fremden ST-Tag durchgeführt werden, so wird ebenfalls anhand der Orientierung die passende Sektor-Warteschlange bestimmt. Das weitere Verfahren geschieht analog zum Vorgehen der einfachen Warteschlangen-Erweiterung, bezogen auf die vorher bestimmte Warteschlange.
Der Vorteil dieser Sektorenaufteilung liegt darin, dass für alle Blickrichtungen historische Informationen zur Verfügung stehen und trotzdem keine unnötig langen Warteschlangen durchlaufen werden müssen. Zudem aktualisieren sich die Warteschlangen asynchron, d.h. solange die Kamera nur in eine Richtung zeigt, wird auch nur die entsprechende Sektorwarteschlange aktualisiert, die anderen bleiben davon unberührt.

Invalidierungsmechanismus Die grundsätzliche Unabhängigkeit der sektorbezogenen Warteschlangen untereinander führt auch zu einem Nachteil: Daten in den Warteschlangen können veralten und zu falsch-positiven Proximitätstests führen. Wenn eine Kamera über einen längeren Zeitraum nur in eine Richtung zeigt, so bleiben in den anderen Warteschlangen alte ST-Tags möglicherweise über längere Zeiträume erhalten. Ein typisches Szenario wäre ein Spaziergang entlang einer geraden, sehr langen Einkaufsstraße. Die Blickrichtungen der Personen schwanken sicher in gewissem Maße zwischen links und rechts der Hauptbewegungsrichtung, werden jedoch nur selten die um 180° gedrehte Ansicht erreichen. Die in diesem Fall „hinten“ liegenden Sektoren sammeln alte ST-Tags an, die an Orten, an denen sich die zugehörige Person schon lange nicht mehr befindet, zu positiven Proximitätstest-Ergebnissen führen können.
Um dieses Problem zu lösen enthält die SWE einen Invalidierungsmechanismus, der immer zur Anwendung kommt, wenn ein neuer ST-Tag in eine Warteschlange eingefügt werden soll. Vor dem Einfügen des neuen ST-Tags wird zunächst wie im ersten Schritt des Basisverfahrens geprüft, ob die Anzahl der im SURF-Merkmalsvektor enthaltenen Merkmalspunkte über dem Grenzwert $\mathrm{th}_{\mathrm{min}}$ liegt. Ist dies nicht der Fall, wird der ST-Tag nicht weiter betrachtet und auch in keine Warteschlange eingefügt.

Wenn die Anzahl der Merkmalspunkte ausreichend ist, so wird anhand der Orientierung des ST-Tags die zugehörige Sektor-Warteschlange bestimmt. Nun wird der ST-Tag mit allen bereits in der Warteschlange enthaltenen ST-Tags verglichen, solange, bis die (analog zum Basisverfahren) berechnete Ähnlichkeit über einem bestimmten Grenzwert liegt. Ist dies bei mindestens einem bereits vorhandenen ST-Tag der Fall, so wird der neue ST-Tag in die Warteschlange eingefügt (und verdrängt evtl. den ältesten daraus). Findet sich jedoch kein ST-Tag in der Warteschlange, zu dem die geforderte Mindestähnlichkeit besteht, so wird ein Fehlerzähler, der für jede Sektor-Warteschlange gespeichert wird, hochgezählt und der neue ST-Tag verworfen, d.h. nicht in die Warteschlange eingefügt.
Erreicht der Fehlerzähler einer Sektor-Warteschlange einen oberen Grenzwert $\mathrm{th}_{\mathrm{Fehler}}$, so wird angenommen, dass sich der Kontext des Benutzers signifikant geändert hat, d.h. der Benutzer sich entweder an einen anderen Ort bewegt hat oder die Umgebung des Benutzers sich verändert hat (z.B. weil sich Personen und Objekte bewegt haben). Die Konsequenz daraus ist, dass angenommen werden muss, dass auch die Informationen in den anderen Sektor-Warteschlangen veraltet sind. Deswegen werden in diesem Fall sämtliche Sektor-Warteschlangen invalidiert, d.h. geleert, und ab diesem Zeitpunkt neu befüllt.
Mit Hilfe dieser Erweiterung sollte in vielen Fällen, in denen Benutzer (länger) in unterschiedliche Richtungen blicken aber trotzdem als „in der Nähe voneinander“ erkannt werden sollen, der Proximitätstest zum korrekt Ergebnis führen, ohne eine zu große Anfälligkeit für falsch-positive Ergebnisse aufgrund veralteter Daten zu entwickeln.

5.5.6 Logik-Ebene

Im Gegensatz zum bereits vorgestellten ProbeTag-Verfahren, das standardmäßig die WLAN-Signale über einen gewissen Zeitraum aggregiert bzw. aggregieren muss, findet beim SURFtogether-Verfahren keine Aggregation bzw. Glättung in dieser Form statt. Die Warteschlangen-Erweiterungen sind keine Aggregation im eigentlichen Sinn, sondern eher eine Historie.
Da bei den visuellen Daten größere Schwankungen wahrscheinlich sind, ist es sinnvoll, auf das bestehende SURFtogether-Verfahren – unabhängig davon, ob und welche Erweiterung verwendet wird – noch eine darüberliegende *Logik-Ebene* anzuwenden, die die berechneten Ähnlichkeiten in gewisser Weise glättet. Diese Logik-Ebene arbeitet folgendermaßen:

- Für den Zeitpunkt t wird für das vorhergehende Zeitfenster $[t - \delta_t, t]$ berechnet, welcher Anteil der darin befindlichen ST-Tags-Vergleiche zu einem Ähnlichkeitswert geführt hat, der als Proximität erkannt wird.

- Liegt dieser Anteil über einem vorher definierten Grenzwert $\mathrm{th}_{\mathrm{ratio}}$, wird für Zeitpunkt t eine Proximität angenommen, andernfalls nicht.

Durch dieses Vorgehen werden Ausreißer in beide Richtungen (falsch-positiv und falsch-negativ) kompensiert und damit das Verfahren gerade in schwierigen Situationen deutlich robuster gemacht.

5.6 Evaluation des SURFtogether-Verfahrens

Das vorgestellte SURFtogether-Verfahren wurde umfangreich evaluiert, sowohl mit Hilfe einer 3D-Simulation als auch mit echten Video-Aufnahmen. Die unterschiedlichen Untersuchungen, Auswertungen und Ergebnisse werden im Folgenden erläutert.

5.6.1 Evaluation mit Hilfe einer Simulation

Um eine Evaluation unter kontrollierten Bedingungen durchführen zu können, wurde eine Simulationsumgebung entwickelt. Es wird in einer 3D-Welt eine kleine städtische Umgebung mit darin umherlaufenden Menschen simuliert. Zusätzlich bewegen sich ein bis vier virtuelle Testpersonen mit am Kopf befestigten Kameras in dieser Welt, wobei kontinuierlich ihr jeweiliges Kamerabild sowie ihre Position aufgezeichnet werden. Zwei der Testpersonen laufen direkt nebeneinander, formen also eine Gruppe mit hoher Proximität. Letzteres gilt sowohl in örtlicher Hinsicht wie auch in Bezug auf die Aktivität, d.h. beide gehen in der selben Geschwindigkeit zum selben Ziel. Die anderen beiden Testpersonen laufen je nach untersuchtem Szenario mehr oder weniger unabhängig von der Gruppe durch die Welt und stellen damit den Gegenpol mit niedriger bzw. keiner Proximität dar.
In Abbildung 5.2 befindet sich ein Screenshot der Simulationsumgebung. Man sieht im Hintergrund eine Vogelperspektive der simulierten Welt sowie in den vier Ecken jeweils das Live-Kamerabild der vier simulierten Testpersonen.

5.6.1.1 Implementierung der Simulationsumgebung und der Auswertungsskripte

Die 3D-Modelle der Stadt bzw. der simulierten Personen stammen aus einem 3D-Modell-Paket des Herstellers Synty Studios [178].
Damit die Personen sich sinnvoll in der 3D-Welt bewegen, wurde ein einfaches Bewegungsmodell implementiert: Jede simulierte Person sucht sich zu Beginn und beim Erreichen eines Zielortes zufällig einen neuen Zielort aus einer Liste von über die gesamte Stadt verteilten POIs. Sie bewegt sich daraufhin auf dem kürzesten, begehbaren Weg zum gewählten Zielort und wählt dann erneut das nächste Ziel. Bei einzelnen untersuchten Situationen, die im weiteren Verlauf dieser Evaluation noch erläutert werden, wurde der Bewegungsraum der simulierten Testpersonen künstlich eingeschränkt, um bestimmte Spezialfälle zu provozieren.

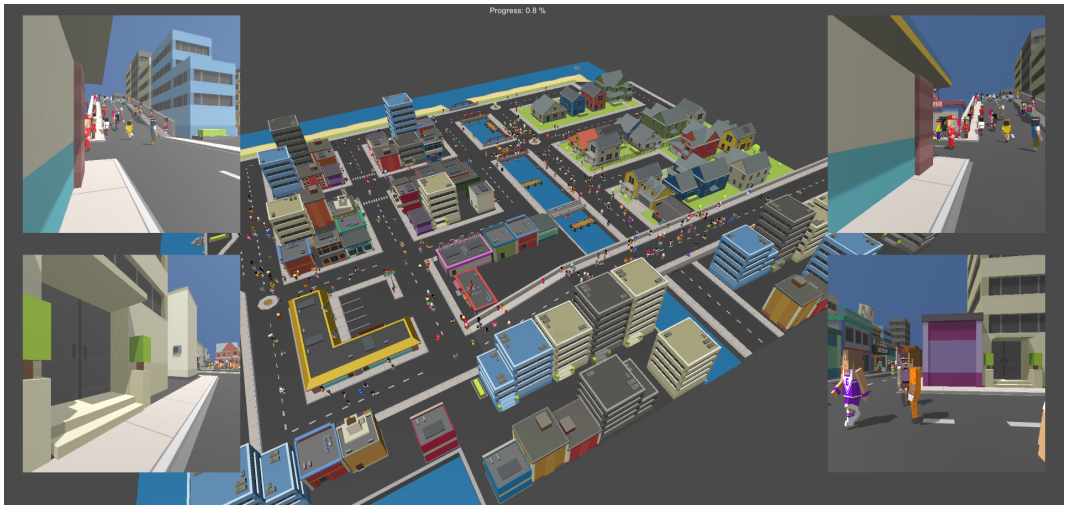


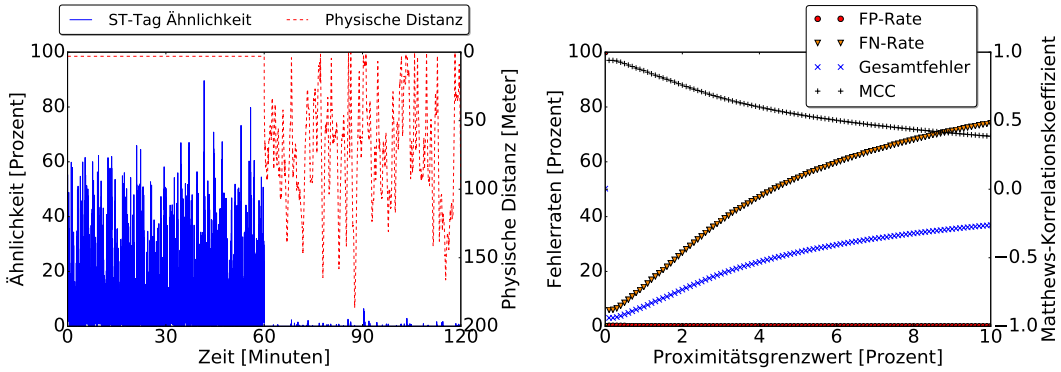
Abbildung 5.2: Screenshot der Simulationsumgebung zur Evaluation des SURFtogether-Verfahrens. Im Hintergrund ist die simulierte Welt aus der Vogelperspektive zu sehen, in den Ecken jeweils überlagert das aktuelle Kamerabild der simulierten Testpersonen.

Neben einer Kamera aus der Vogelperspektive zur Überwachung der Simulation existieren ein bis vier Kameras, die an den bis zu vier simulierten Testpersonen angebracht sind. Die Ausgabe dieser Kameras wird zur Kontrolle live im Benutzerinterface angezeigt. Zudem wird mit einer Frequenz von 10 Bildern pro Sekunde und einer Auflösung von 720x480 Pixeln das aktuelle Kamerabild persistent gespeichert, angereichert mit Meta-Daten, insbesondere der aktuellen Position der Kamera (d.h. der Person) im Koordinatensystem der 3D-Welt. Die Simulation dient rein zur Erzeugung der (Video-)Datensätze, die anschließend reproduzierbar ausgewertet werden können.
Die Anwendung des SURFtogether-Verfahren erfolgt unabhängig von der Simulationsumgebung in einer gesonderten Implementierung. Dabei wird die OpenSURF-Implementierung des SURF-Algorithmus [48] unter Verwendung eines Java-Wrappers [174] eingesetzt.

5.6.1.2 Statische Blickrichtung

Zunächst wurde zur Überprüfung der grundsätzlichen Funktionsweise des SURFtogether-Verfahrens eine Art Basis-Szenario untersucht. Hierbei bewegen sich zwei Testpersonen gemeinsam in festem Abstand von 3 Metern durch die simulierte 3D-Welt. Sie bilden also eine Gruppe mit hoher Proximität, die es durch das Verfahren zu erkennen gilt. Nach einiger Zeit wird diese Verbindung aufgelöst und die beiden Personen bewegen sich getrennt voneinander durch die 3D-Welt. In dieser Phase sollte also möglichst keine Ähnlichkeit mehr erkannt werden.
In diesem Szenario wurde zudem eine erleichternde Einschränkung eingeführt.

Die beiden Testpersonen (bzw. die simulierten Kameras) schauen – in der Phase, in der sie eine Gruppe bilden – grundsätzlich in die gleiche Richtung, d.h. in die gemeinsame Bewegungsrichtung. Die Kamerabilder unterscheiden sich dennoch. Zum einen wird allein durch den örtlichen Versatz der beiden Personen ein etwas anderer Bildausschnitt aufgenommen, zum anderen führen durch das Bild laufende fremde Personen jeweils zu unterschiedlichen Verdeckungen. Ausgeschlossen wurden dadurch aber noch größere Unterschiede, die durch – in realen Situation durchaus gegebenen – Kopfbewegungen der Testpersonen verursacht werden.
In diesem Szenario wurde zunächst ein Zeitraum von einer Stunde simuliert, in dem die beiden Personen zusammen unterwegs waren, und anschließend ein weiterer Zeitraum dieser Länge, in dem sie sich unabhängig voneinander fortbewegten. Wendet man nun das in Kapitel 5.5.3 erläuterte Basisverfahren an, ergibt sich der in Abbildung 5.3a dargestellte Ähnlichkeitsverlauf.



(a) Ähnlichkeiten und räumliche Distanzen über die Zeit

(b) Auswirkungen unterschiedlicher Proximitätsgrenzwerte

Abbildung 5.3: Ähnlichkeitsverlauf und Fehlerraten in einem Basis-Szenario ohne Kopfbewegung der Testpersonen.

An der Abszisse ist hier der zeitliche Verlauf des Experiments angetragen, auf den beiden Hochachsen findet sich zum einen die berechnete Ähnlichkeit und zum anderen die tatsächlich physische Distanz (mit inverter Achse!), die anhand der Ortsinformationen aus der Simulation exakt berechnet werden konnte. In blau ist die nach dem Basisverfahren berechnete Ähnlichkeit zum jeweiligen Zeitpunkt dargestellt, in rot der Verlauf der physischen Distanz.
Wie für das Szenario definiert, befinden sich die beiden Testpersonen in der ersten Hälfte des Experiments durchweg im gleichen, sehr kurzen Abstand. In der zweiten Hälfte variiert die Distanz wie gewollt, da sich beide Testpersonen unabhängig voneinander fortbewegt haben. Hier kann es natürlich auch immer wieder zu (Beinahe)-Begegnungen kommen, wie die teilweise sehr kurzen Distanzen zeigen, jedoch formen die beiden Testpersonen in dieser zweiten Phase zu keinem Zeitpunkt mehr eine Gruppe, d.h. ihre Aktivität im dem Sinne, dass sie sich gemeinsam auf das gleiche Ziel zu bewegen, ist unterschiedlich.

Betrachtet man nun den Verlauf der Ähnlichkeitswerte, so lassen sich die beiden Phasen des Experiments visuell unterscheiden. In der ersten Hälfte bestehen zwar große Schwankungen in der berechneten Ähnlichkeit, aber sie erreicht immer wieder, in kurzen Abständen, hohe und sehr hohe Werte bis hin zu fast 90%. In der zweiten Hälfte tritt dagegen fast gar keine nenneswerte Ähnlichkeit auf, insbesondere auch nicht zu den Zeitpunkten, an denen ansich eine hohe physische Nähe durch Zufall gegeben war. Dies deutet also bereits, wie im Konzept auch angedacht, darauf hin, dass für das SURFtogether-Verfahren implizit auch die Aktivität der Testpersonen, d.h. ob diese übereinstimmt oder nicht, eine große Rolle für die vermutete Proximität spielt.
Es stellt sich nach der visuellen Betrachtung nun die Frage, ob mit dem Basisverfahren in diesem Fall auch eine konkrete Aussage über den Proximitätsstatus der zwei Testpersonen gemacht werden kann. Hierzu wurde untersucht, inwiefern sich verschiedene Grenzwerte, die man zur Unterscheidung von Phasen mit Proximität und solchen ohne verwenden kann, auf die Zuverlässigkeit der Aussage auswirken. Die Ergebnisse sind in Abbildung 5.3b dargestellt.
An der Abszisse sind hier nun verschiedene Grenzwerte aufgetragen, dargestellt in Prozent (analog zur linken Hochachse der vorherigen Abbildung 5.3a). In Richtung der Ordinate findet sich der prozentuale Wert für verschiedene Fehlerarten.
Die beiden roten Kurven zeigen die prozentualen Werte der beiden Fehlerraten falsch positiv (rote Kreise) und falsch negativ (rote Dreiecke). Ein „falsch positives“ Ergebnis entspricht hierbei einer positiven Proximitätseinschätzung, obwohl gar keine Proximität gegeben war. Im Graphen dargestellt ist in dem Fall das Verhältnis aus falsch positiven Einschätzungen zur Gesamtzahl der Zeitpunkte, an denen keine Proximität gegeben war. Ein „falsch negatives“ Ergebnis bedeutet, dass die Proximitätseinschätzung des Verfahrens negativ ausfällt, obwohl tatsächlich eine Proximitätssituation gegeben war. Der dargestellte Wert bezieht sich in diesem Fall auf das Verhältnis aus falsch negativen Ergebnissen zur Gesamtzahl der Situationen, in denen eine Proximität hätte erkannt werden müssen.
In blau (Kreuze) ist der Gesamtfehler dargestellt, d.h. die Anzahl der falsch eingeschätzten Situationen (egal in welcher Form) im Verhältnis zu allen Situationen. Zudem ist in schwarz (Plus-Zeichen) der sogenannte *Matthews-Korrelations-Koeffizient (MCC)* [**?**] angetragen. Dieser ist definiert als

$$\mathrm{MCC} = \frac{\mathrm{TP} \cdot \mathrm{TN} - \mathrm{FP} \cdot \mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}} \tag{5.8}$$

wobei TP die Anzahl richtig positiver, TN die Anzahl richtig negativer, FP die Anzahl falsch positiver und FN die Anzahl falsch negativer Einschätzungen bedeutet. Der MCC nimmt Werte zwischen 1 (perfekte Vorhersage) und -1 (komplett falsche Vorhersage) an, wobei 0 einem zufälligen Raten der Einschätzung entspricht. Ziel sind also möglichst hohe Werte. Der MCC bietet neben einer Aggregation der Falsch-Positiv- und Falsch-Negativ-Rate auch den

Vorteil, dass er unabhängig von der Verteilung der Proximitäts- und Nicht-Proximitätsfälle ist. In den Simulationen sind durch die kontrollierten Bedingungen die beiden Fälle zwar sowieso gleichmäßig aufgeteilt, in den späteren realen Versuchen kann diese Ausgewogenheit jedoch nicht mehr hundertprozentig sichergestellt werden.
In der Grafik sind die unterschiedlichen Grenzwerte in Richtung der Abszisse in Abstufungen von 0,1% dargestellt. Feinere Abstufungen, insbesondere auch im Bereich unter 0,1%, haben zu keinen nennenswerten Zusatzerkenntnissen geführt und sind zudem auch für Vergleich mit den später betrachteten Szenarien zu feingranular.
Betrachtet man die Verläufe, ist zunächst wichtig festzustellen, dass bei einem Proximitätsgrenzwert von 0% alle Situationen als „keine Proximität vorhanden“ eingeschätzt werden, wodurch keine falsch postiven Ergebnisse auftreten können, aber eben auch keine richtig positiven. Dies ist natürlich keine sinnvolle Wahl eines Grenzwertes. Sinnvolle Belegungen beginnen in diesem Fall ungefähr bei 0,1%.
Überraschend ist zunächst festzustellen, dass es in diesem Szenario im relevanten Bereich der Grenzwerte durchweg zu so gut wie keinen falsch positiven Einschätzungen kommt (maximal 0.2%). Letztendlich steht dies aber mit der visuellen Einschätzung des Ähnlichkeitverlaufs über die Zeit (vgl. Abbildung 5.3a) im Einklang. In der Phase ohne Proximität wird tatsächlich so gut wie keine Ähnlichkeit zwischen den Testpersonen festgestellt.
Mit zunehmendem Proximitätsgrenzwert wird daher zunächst nur die Falsch-Negativ-Rate beeinflusst. Setzt man den Grenzwert höher an, so werden mehr Situationen mit kleineren Ähnlichkeitswerten, in denen aber eine Proximitätssituation gegeben war, falsch eingeschätzt. Diese Steigerung der falsch negativen Einschätzungen ist letztendlich dann auch der Grund für den steigenden Gesamtfehler bzw. sinkenden MCC.
Mit einem MCC von 0.94 bei einem Ähnlichkeits-Grenzwert von 0.1 lässt sich erkennen, dass das Verfahren grundsätzlich funktioniert.

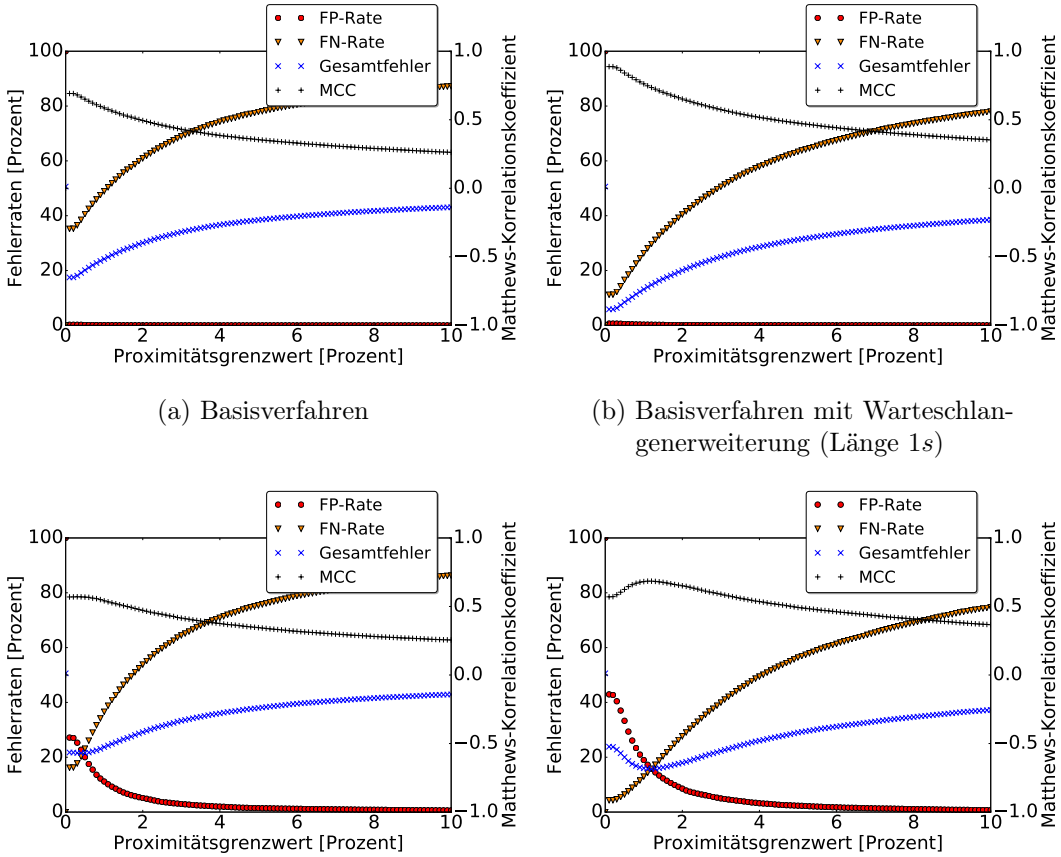
5.6.1.3 Dynamische Kopfbewegung

Im ersten Testszenario wurden bewusst Kopfbewegungen der Testpersonen ausgeschlossen, um das Basisverfahren ohne diesen Einflussfaktor zu beurteilen. Es hat sich gezeigt, dass unter diesen erleichterten Bedingungen, das Basisverfahren alleine schon das Potential aufweist, zuverlässige Proximitätseinschätzungen zu liefern.
In der Realität ist die Blickrichtung einer Person jedoch nicht in diesem Maße starr. Je nach Trageposition der Kamera – am Kopf mehr, vor der Brust weniger – schwankt der Bildausschnitt durch die Bewegung des Körpers.
Um dieses Verhalten zu berücksichtigen, wurde die Simulation um eine entsprechende Bewegungsfreiheit der Kameras erweitert. Die Rotation der Kamera einer Person um die y-Achse des Koordinatensystems der 3D-Welt – in diesem Fall der Hochachse – wird dazu in regelmäßigen Abständen verändert,

d.h. die Blickrichtung schwankt links und rechts der eigentlichen Bewegungsrichtung hin und her. Eine Bewegung nach oben und/oder unten wurde nicht berücksichtigt. Die Kamera rotiert nach folgendem Muster:

- Es wird zufällig ein Rotationsziel aus dem Intervall $[-180°, +180°]$ gezogen (relativ zur aktuellen Bewegungsrichtung, die als 0° aufgefasst wird).
- Nur wird mit einer fixen Rotationsgeschwindigkeit ($\frac{60°}{s}$) die Blickrichtung der Kamera hin zu dem gewählten Rotationsziel transformiert.
- Ist das Ziel erreicht, wird erneut ein zufälliges Ziel gezogen, dieses Mal jedoch aus dem Intervall $[-15°, +15°]$. Das nächste Rotationsziel entspricht also ungefähr wieder der Bewegungsrichtung ($\pm 15°$).
- Es wird erneut zum Ziel rotiert. Bei Erreichen des Ziels beginnt der Ablauf von vorne.

Die Blickrichtung kann sich also in jede Richtung um bis zu 180° ändern, schwenkt aber auch immer wieder grob zur eigentlichen Bewegungsrichtung zurück. In der Realität ist der Blick auch primär nach vorne gerichtet, wenn man sich fortbewegt.
Das so konfigurierte und ansonsten zum ersten Test identische Szenario wurde erneut zwei Stunden simuliert, die erste Hälft wieder in einer Gruppensituation, die zweite als Fortbewegung zweier Testpersonen unabhängig voneinander. In Abbildung 5.4 sind erneut im gleichen Stil wie vorher in Abbildung 5.3b die Auswirkungen verschiedener Proximitätsgrenzwerte auf die Fehlerraten dargestellt. Es sind dabei bereits auch die Ergebnisse der erweiterten Verfahren mit Warteschlange und sektorbezogener Warteschlange enthalten, die im Folgenden ebenso erläutert werden.
Beim Basisverfahren (vgl. Abbildung 5.4a) ist grundsätzlich zu beobachten, dass kein so niedriger Gesamtfehler wie noch beim Szenario ohne Kopfbewegung erzielt werden kann (17.5% im Vergleich zu 3.1% vorher). Es entstehen zwar wieder so gut wie keine falsch positiven Einschätzungen (maximal 0.2%), jedoch steigt die Zahl der falsch negativen Ergebnisse (im besten Fall 35.2% im Vergleich zu 6.0% ohne Kopfbewegung) und damit der Gesamtfehler deutlich. Dies entspricht im Prinzip der bereits im konzeptuellen Teil beschriebenen Erwartung, dass das Umherschauen der Personen bei den 1-zu-1-Vergleichen des Basisverfahrens zu Fehleinschätzungen führen kann, allein dadurch, dass in diesem Fall auch in einer Proximitätssituation unterschiedliche Bilder entstehen.
Aus diesem Grund wurden mehrere Erweiterungen des Basisverfahrens vorgeschlagen (vgl. Kapitel 5.5). Die erste ist die Warteschlangen-Erweiterung, deren Grundidee darin besteht, in gewissem Maße ältere Bilder vorzuhalten und in die Ähnlichkeitsberechnung mit einzubeziehen. Primär besteht das Ziel dieser Erweiterung darin, kurzzeitige „Probleme“ im Bild, z.B. ein durch ein Objekt verdeckter Bildausschnitt, zu umgehen. Im gleichen Zuge kann die Erweiterung



(a) Basisverfahren

(b) Basisverfahren mit Warteschlangenerweiterung (Länge 1s)

(c) Basisverfahren mit sektorbezogener Warteschlangenerweiterung (9 Sektoren, 3 Bilder pro Sektor)

(d) Basisverfahren mit sektorbezogener Warteschlangenerweiterung (9 Sektoren, 6 Bilder pro Sektor)

Abbildung 5.4: Fehlerraten für verschiedene Konfigurationen des SURFtogether-Verfahrens in einem Szenario mit dynamischer Kopfbewegung.

aber möglichweise Blickrichtungsschwankungen durch die Bildhistorie aus der Warteschlange ausgleichen.

Dies zeigt sich auch in den Ergebnissen dieses Testdurchlaufs (vgl. Abbildung 5.4b). Die Warteschlangen-Erweiterung schafft es, die Zahl der falsch negativen Einschätzungen deutlich zu senken (11.2% bei Verwendung des Grenzwertes mit dem niedrigsten Gesamtfehler). Zudem erhöht sich die Zahl der falsch positiven Einschätzungen nur unwesentlich (0.6%). Zusammengenommen führt dies zu einem deutlich reduzierteren Gesamtfehler (5.8%), d.h. die Warteschlangen-Erweiterung bewirkt tatsächlich eine Verbesserung bei dynamischen Kamerabewegungen.

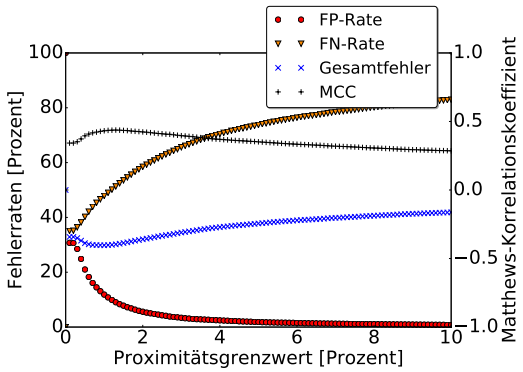
Für genau die Problemstellung der wechselnden Blickrichtung wurden die sektorbezogenen Warteschlangen konzipiert. Bei dieser Erweiterung existiert nicht nur eine Warteschlange, sondern mehrere dediziert für verschiedene Orientie-

rungsrichtungen. In Abbildung 5.4c sind die Ergebnisse einer Konfiguration mit 9 Sektoren und einer Warteschlangengröße von drei bzw. sechs Bildern je Sektor dargestellt. Wie beabsichtigt führt die Verwendung dieser Erweiterung zu einer Verringerung der falsch negativen Proximitätseinschätzungen (im besten Fall 16.2%). Gleichzeitig treten in diesem Fall aber vermehrt falsch positive Einschätzungen auf, sodass im Endeffekt keine Verbesserung des Gesamtfehlers erreicht werden kann (im besten Fall 21.5%).
Etwas besser sieht das Ergebnis bei Verwendung von größeren Sektorwarteschlangen aus, wie in Abbildung 5.4d zu sehen ist. Mit einer Konfiguration von erneut 9 Sektoren und nun jedoch 6 Bildern pro Warteschlange kann die Zahl der falsch negativen Einschätzung in so großem Maße gesenkt werden, dass das vermehrte Auftreten von falsch positiven Ergebnissen kompensiert und ein im Vergleich zum Basisverfahren geringerer Gesamtfehler (15.8%) möglich ist.
Mit 9 Bildern pro Warteschlange und einer Bildrate von 10 Bildern pro Sekunde erreichen bei letzterer Konfiguration die sektorbezogenen Warteschlangen jedoch nahezu die Länge der einfachen Warteschlangen-Erweiterung, sodass die angestrebten Performanz-Vorteile der Sektorwarteschlangen nicht realisierbar sind. Gleichzeitig ist der niedrigste Gesamtfehler in diesem Testszenario mit der einfachen Warteschlangenerweiterung möglich. Die sektorbezogenen Warteschlangen erweisen sich in diesem Fall nur dann als vorteilhaft, wenn die Zahl der falsch negativen Einschätzungen optimiert werden soll.
Mit einem MCC von 0.89 im besten Fall mit der Warteschlangenerweiterung ist jedoch auch in diesem Szenario zu sehen, dass das SURFtogether-Verfahren grundsätzlich funktionieren kann.

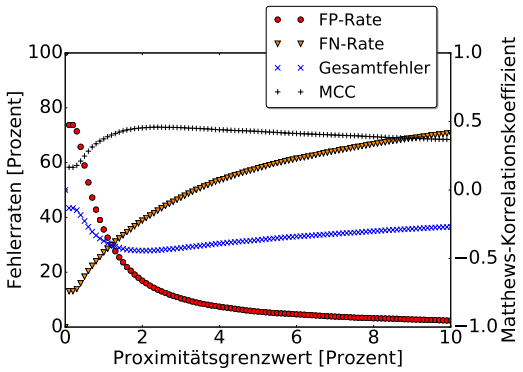
5.6.1.4 Einfluss der Aktivität / Gruppensituation

Bereits im ersten Testszenario konnte beobachtet werden, dass die in der zweiten Phase (ohne gezielte Proximitätssituationen) zufällig entstandenen kurzen Distanzen zwischen den Testpersonen kaum zu falschen Einschätzungen geführt haben (vgl. Kapitel 5.6.1.2). Wie bei der Vorstellung des SURFtogether-Verfahrens erläutert, besteht die Idee dieses Ansatzes darin, dass die Proximitätssituation nicht nur durch eine örtliche Nähe sondern auch durch die Ähnlichkeit der Aktivität definiert ist, letztere daher einen Einfluss auf die berechnete Proximitätsschätzung haben sollte.
Um nun dieses Ziel und die eingangs beobachteten ersten Anzeichen genauer zu untersuchen, wurde ein weiteres Testszenario definiert. In diesem Fall bewegen sich erneut zwei Testpersonen miteinander in einer Gruppe durch die 3D-Welt, befinden sich also in einer zu erkennenden Proximitätssituation. Gleichzeitig wird eine dritte Testperson instanziiert, die sich unabhängig von den beiden anderen durch die 3D-Welt bewegen soll.
Zur Überprüfung des Einfluss der gemeinsamen Aktivität, d.h. dass die ersten beiden Testpersonen zusammen unterwegs sind und die dritte sich von ihnen unabhängig bewegt, wurde für alle drei Personen ein maximaler Bewegungsradius definiert. Die Beschränkung wurde so gewählt, dass sich alle drei Personen

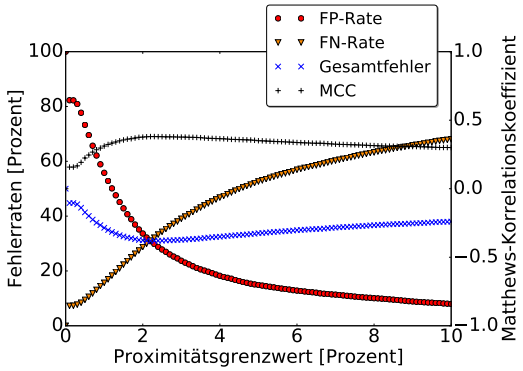
nur auf einem zentralen Platz sowie den ersten Metern der angrenzenden Straßen der simulierten 3D Welt bewegen. Somit hielten sich die Testpersonen also durchgehend am selben Platz in der 3D-Welt auf.



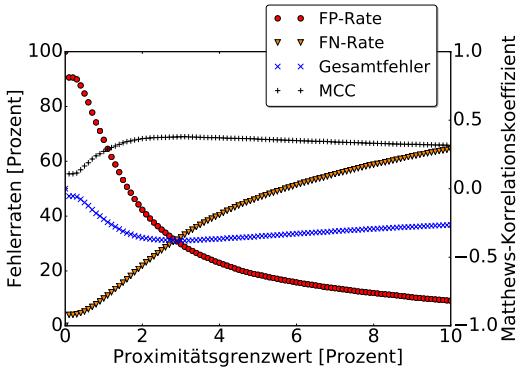
(a) Basisverfahren



(b) Basisverfahren mit Warteschlangenerweiterung (Länge $1s$)



(c) Basisverfahren mit sektorbezogener Warteschlangenerweiterung (9 Sektoren, 3 Bilder pro Sektor)



(d) Basisverfahren mit sektorbezogener Warteschlangenerweiterung (9 Sektoren, 6 Bilder pro Sektor)

Abbildung 5.5: Fehlerraten für verschiedene Konfigurationen des SURFtogether-Verfahrens in einem Szenario, indem sich sowohl Benutzer, die zur selben Gruppe gehören, als auch Benutzer, die davon unabhängig sind, in großer räumlicher Nähe befinden.

Betrachtet man Proximität also rein ortsbasiert, so sind alle drei Testpersonen in vielen Fällen als „nah beieinander“ einzustufen. Wie am Anfang dieser Arbeit beschrieben (vgl. Kapitel 2.4.2), ist statt der rein örtlichen Nähe vielmehr auch eine gruppenbezogene Proximität von Interesse. Somit ist das gewünschte Ergebnis, dass eben doch nur die zwei ersten Testpersonen als zusammengehörend erkannt werden, trotz der räumlichen Nähe aller drei Testpersonen.
Dieses Szenario wurde 60 Minuten lang simuliert, wobei die Testpersonen eins und zwei durchgehend zusammen unterwegs waren und die dritte Testperson

von ihnen unabhängig. Es gab in diesem Fall keine Phase, in der die ersten beiden Testpersonen getrennt waren. Die Ergebnisse sind in Abbildung 5.5 visualisiert.
Allgemein ist zu erkennen, dass dieses sehr eng gesteckte Szenario wie erwartet eine höhere Anfälligkeit für falsch positive Einschätzungen mit sich bringt. Dadurch, dass sich die Testpersonen in sehr großer räumlicher Nähe befinden, basiert die Einschätzung der (kontextuellen) Proximität implizit zu großen Teilen auf der gemeinsamen bzw. unterschiedlichen Aktivität. Dieser Versuch zeigt, dass der Proximitätsgrenzwert zur Beseitigung einer großen Menge an falsch positiven Einschätzung etwas höher gewählt sollte als bisher angenommen. Bei Verwendung der Warteschlangen-Erweiterung kann bei Verwendung eines Grenzwertes von 2.0% immerhin ein MCC von 0.46 erreicht werden. Dies ist im Hinblick auf das schwierige Szenario ein sehr gutes Ergebnis und insbesondere deutlich besser als eine zufällige Abschätzung.

5.6.1.5 Beschränkung auf dynamische Objekte

In den bisher beschriebenen Testszenarien konnte gezeigt werden, dass eine grundsätzliche Proximitätserkennung mit Hilfe des SURFtogether-Verfahrens umsetzbar ist. Wie in Abschnitt 5.4 beschrieben, existieren einige Ansätze, die versuchen, anhand von vorher aufgenommenen Bild-Datenbanken eine Positionsbestimmung über Bildvergleiche vorzunehmen. Mit Hilfe solcher Datenbanken könnte wiederum die Privatsphäre der Benutzer bei Verwendung des SURFtogether-Verfahrens untergraben werden. Die übermittelten ST-Tags besitzen zwar keine tatsächliche Positionssemantik, durch einen Abgleich in der beschriebenen Form könnte die Position eines Benutzers jedoch trotzdem abgeleitet werden, wenn eine entsprechende Datenbank existiert.
Wie im vorherigen Abschnitt gezeigt, kann das SURFtogether-Verfahren relativ gut zwischen Personen, die sich nur am selben Ort befinden, und Personen, die sich tatsächlich in einer Gruppe befinden, unterscheiden. Daraus kann gefolgert werden, dass die Möglichkeiten zur Manipulation eines Proximitätstests mit Hilfe einer Datenbank gering sind. Problematisch erscheint also nur das Privatsphäre-Thema an sich.
Ähnlich wie beim ProbeTag-Verfahren lässt sich dieses Problem in Kombination mit einem PPT-Verfahren lösen, das sicherstellt, dass die originalen ST-Tags von keiner anderen Partei gelesen werden können. Darüberhinaus kann man jedoch auch versuchen, das Bildmaterial lokal zu bewerten, um statische von dynamischen Objekten zu unterscheiden, und nur letztere zur Konstruktion der ST-Tags heranzuziehen. Damit wären keine Vergleiche mit einer Datenbank mehr möglich.
Eine Erkennung und Extraktion von dynamischen Objekten kann beispielsweise mit Hilfe von Algorithmen zur Hintergrundsubtraktion [**?**] durchgeführt werden. Für eine weitere Evaluation wird daher angenommen, dass ein entsprechendes Verfahren existiert. Um herauszufinden, ob auf einem in dieser Art und Weise reduzierten Bildmaterial das SURFtogether-Verfahren nach wie

vor angewendet werden kann, wurde eine weitere Simulation durchgeführt, in der sämtliche Gebäude aus der 3D-Welt entfernt wurden, sodass nurmehr der (generische) Untergrund und die sich bewegenden Menschen übrig waren. In Abbildung 5.6 ist ein Screenshot der entsprechend modifizierten Simulation zu sehen.

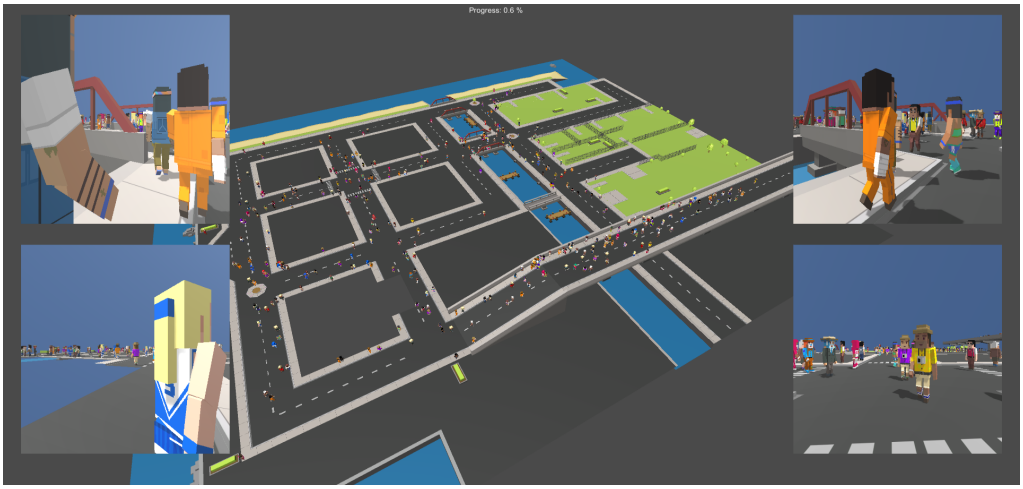
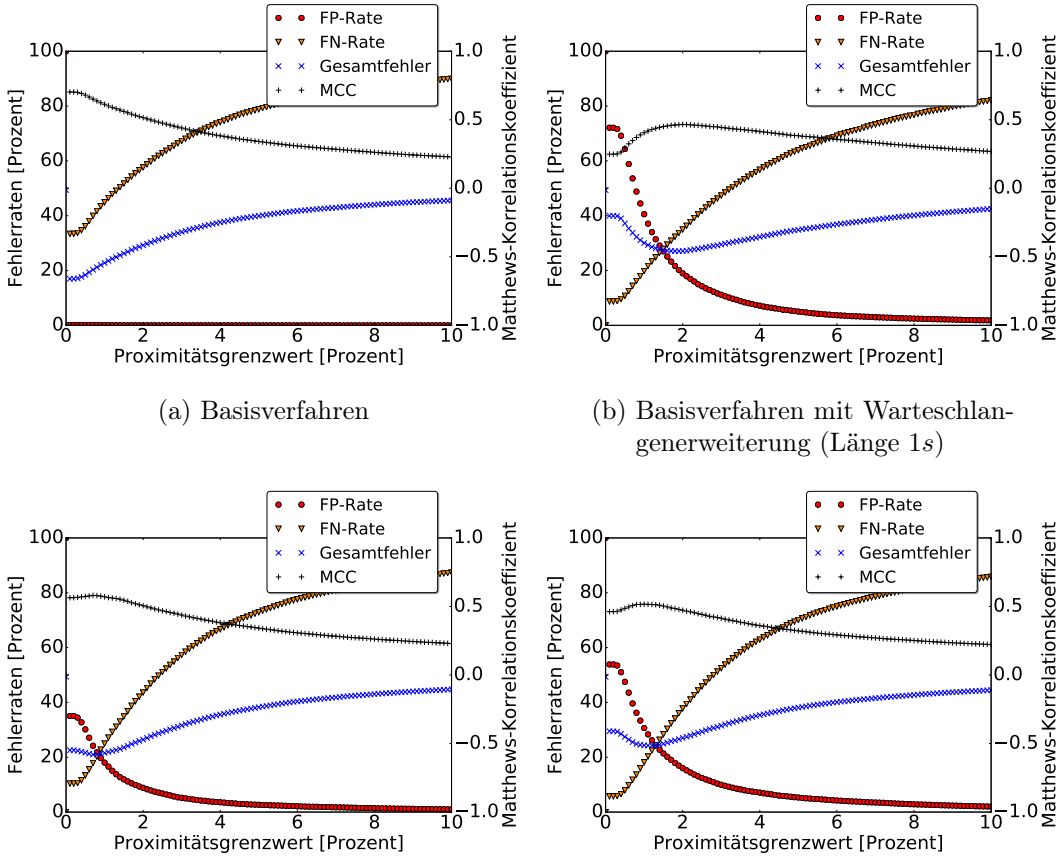


Abbildung 5.6: Simulationsumgebung für ein Szenario ohne statische Objekte.

Die Fehlerraten der verschiedenen SURFtogether-Varianten sind in Abbildung 5.7 dargestellt. Im Allgemeinen entsprechen die Ergebnisse den Werten der vorhergehenden Szenarien, bis auf eine erhöhte Anfälligkeit für falsch positive Einschätzung bei der Verwendung von Warteschlangen. Dass diese Anfälligkeit in diesem Szenario größer ausfällt als bei den anderen Szenarien ist mit dem eingeschränkten Bildmaterial zu erklären, das zum einen durch die entfernten Gebäude, zum anderen durch die 3D-Welt selbst zu erklären ist: Die verwendeten Personen-Modelle sind teilweise Duplikate um die gewünschte Anzahl an simulierten Menschen zu erreichen. Dies kann zu Verwechslungen führen, die in einer vielfältigeren (echten) Umgebung nicht auftreten sollten.
Trotz dieser Problematik können auch in diesem Szenario MCC-Werte über 0.5 erreicht werden, was ein gutes Ergebnis darstellt.

5.6.2 Evaluation in einem realen Szenario

Die bisher beschriebenen Evaluationen wurden alle mit Hilfe der eigens entwickelten Simulationsumgebung durchgeführt. Die Verwendung der Simulation bietet viele Vorteile. So lassen sich damit die Testszenarien sehr genau definieren und umsetzen, Entfernungen zwischen den Testpersonen können exakt gemessen und eingehalten werden, es kann eine beliebige Umgebung als 3D-Modell verwendet werden und die Testpersonen können sich darin frei bewegen. Die bisherigen Ergebnisse aus diesen Testreihen zeigen auch, dass das SURFtogether-Verfahren grundsätzlich als Proximitätserkennungsverfahren eingesetzt werden kann.



(a) Basisverfahren

(b) Basisverfahren mit Warteschlangenerweiterung (Länge 1s)

(c) Basisverfahren mit sektorbezogener Warteschlangenerweiterung (9 Sektoren, 3 Bilder pro Sektor)

(d) Basisverfahren mit sektorbezogener Warteschlangenerweiterung (9 Sektoren, 6 Bilder pro Sektor)

Abbildung 5.7: Fehlerraten für verschiedene Konfigurationen des SURFtogether-Verfahrens in einem Szenario, indem sämtliche statische Objekte entfernt wurden.

Gerade bei der Verwendung von visuellen Informationen stellt sich die Frage, ob die gewonnen Erkenntnisse auch auf die „echte“ Welt übertragbar sind. Eine realitätsgetreue Darstellung von 3D-Umgebungen ist nach wie vor eine große Herausforderung und wurde bei der verwendeten Simulation auch bewusst gar nicht erst versucht. Stattdessen sollen in einem weiteren Testlauf nun tatsächlich in der realen Welt aufgezeichnete Daten zur Auswertung herangezogen werden.

Der im Folgenden evaluierte Testdatensatz wurden von Klette in seiner, vom Autor dieser Arbeit betreuten, Abschlussarbeit [103] aufgezeichnet, und im Rahmen der vorliegenden Arbeit vom Autor (erneut) evaluiert.

5.6.2.1 Aufzeichnung der Testvideos und Sensordaten

Die Datenaufzeichnung der realen Szenarien wurde mit Hilfe von Android-Smartphones [70] vom Typ Nexus 4 [73] des Herstellers LG [116] durchgeführt. Die Verwendung von Smartphones entspricht nicht vollständig der im Konzept eigentlich angedachten Geräteklasse, kann jedoch bei geeigneter Positionierung als passender Ersatz betrachtet werden. Jede Testperson hat zur Datenaufzeichnung das Smartphone in der Hand ungefähr auf Höhe der Brust und die Kamera nach vorne ausgerichtet getragen. Die Schwankungen der Blickrichtung sollten daher ein Mittelmaß aus einem am Kopf getragenen Gerät und einem an der Brust angebrachten Gerät sein.
Auf den Smartphones wurde eine dedizierte Logging-Applikation installiert, die neben den Bilddaten zusätzlich auch noch die zugehörigen Meta-Daten wie die GPS-Position und die Orientierung des Geräts aufgezeichnet hat. Zudem konnte mitgeloggt werden, mit welchen Personen man sich gerade in einer Gruppe befindet.

5.6.2.2 Versuchsablauf

Um einen möglichst realistischen und viele verschiedene Situationen umfassenden Datensatz zu erhalten, wurde ein umfangreiches Test-Szenario skizziert und von zwei Testpersonen durchlaufen. Die Testumgebung war der Münchner Innenstadtbereich rund um das Universitätshauptgebäude am Geschwister-Scholl-Platz 1. Es wurden sowohl Innenbereiche (im Universitätshauptgebäude) als auch Außenbereiche durchlaufen.
Der Ablauf wurde vorher so definiert, dass annähernd gleich viele Phasen, in denen sich die beiden Testpersonen zusammen als Gruppe fortbewegt haben, und Phasen, in denen sie unabhängig voneinander unterwegs waren, aufgezeichnet werden konnten. Der Ablauf bestand aus folgenden Abschnitten:

- Die beiden Testpersonen starten gemeinsam im Universitätshauptgebäude und bewegen sich im Innenraum zusammen durch die gleichen Gänge. In dieser Phase trennen sie sich einmal kurz, um auf unterschiedlichen Wegen zum nächsten Treffpunkt zu gelangen.

- Nach einem weiteren gemeinsamen Weg durch das Gebäude trennen sich die Testpersonen, um parallel durch verschiedene, aber im Wesentlichen baugleiche Gänge des Gebäudes zu laufen.

- Die Personen treffen sich wieder an einem vereinbarten Punkt und gehen gemeinsam ins Freie.

- Nach einer gemeinsamen Phase im Freien begibt sich eine Person auf die anderen Straßenseite, um so zwar ungefähr am gleichen Ort in die gleiche Richtung, aber nicht mehr in der Gruppe zu laufen.

- Nach einer gemeinsamen Phase wechselt eine Person die Straßenseite indem sie durch eine Unterführung läuft. Im Anschluss laufen die beiden Personen ungefähr parallel aber auf unterschiedlichen Straßenseiten (also nicht in einer Gruppe) bis zu einem großen, sehr belebten Platz.

- Auf dem Platz treffen sich die Testpersonen wieder und spazieren gemeinsam über diesen.

- Nach einiger Zeit trennen sich die Personen und laufen individuell über den Platz, sodass sie wiederum ungefähr am gleichen Ort, jedoch nicht zusammen sind.

- Zum Abschluss treffen sich die Personen wieder und verlassen gemeinsam den Platz.

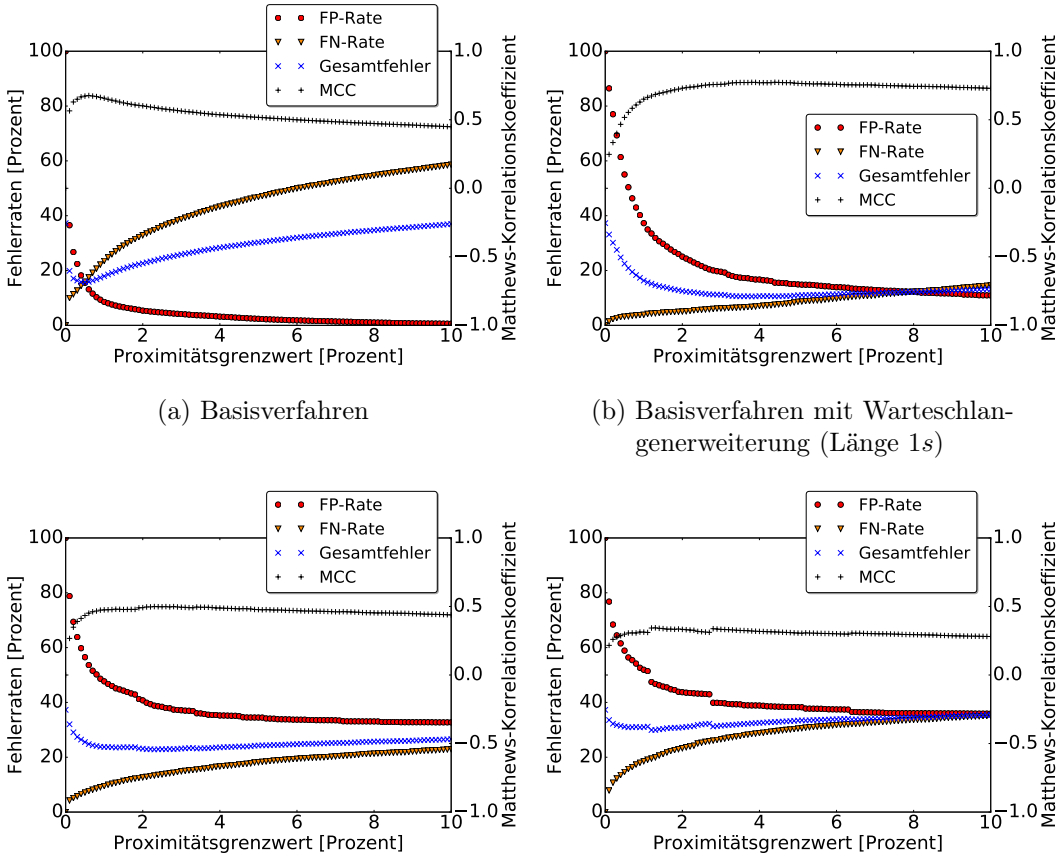
Es wurden also auch verschiedene Situationen erzwungen, die für das SURFtogether-Verfahren schwer zu klassifizieren sein könnten.

5.6.2.3 Ergebnisse im realen Testszenario

Die Ergebnisse dieses Versuchs sind in Abbildung 5.8 dargestellt.
Sowohl das Basisverfahren alleine als auch insbesondere das Verfahren mit einer Warteschlange weisen auch in diesem realen Szenario sehr gute Ergebnisse auf. Letztere Variante erreicht einen maximalen MCC-Wert von 0.77, mit dem in den vorherigen Auswertungen als sinnvoll identifizierten Grenzwert von 2.0% auch einen sehr guten Wert von 0.73.
Die beiden Varianten mit sektorbezogenen Warteschlangen schneiden deutlich schlechter ab. Hier macht sich allerdings eine technische Schwierigkeit bemerkbar: Der Kompass der verwendeten Smartphones arbeitet nicht exakt und stabil genug, um sinnvolle Zuordnungen zu Sektoren vornehmen zu können. Dadurch kommen die intendierten Vorteile dieser Erweiterung nicht zum Tragen.
Insgesamt zeigen diese Ergebnisse, das das SURFtogether-Verfahren auch mit realen Videoaufzeichnungen funktioniert und daher als Proximitätserkennungsverfahren geeignet ist.

5.6.3 Logik-Ebene

Die bisherigen Evaluationen beziehen sich auf unterschiedliche Konfigurationen des Verfahrens. Wie in Abschnitt 5.5.6 beschrieben, erscheint es sinnvoll, den zeitlichen Verlauf der Ähnlichkeiten zu betrachten und mit Hilfe einer zusätzlichen Logik-Ebene die abgeleiteten Proximitätseinschätzungen zu verbessern bzw. robuster zu machen.
Im Folgenden sind beispielhaft für alle Szenarien die Ergebnisse aus dem standardmäßigen Simulationsszenario (vgl. Abschnitt 5.6.1.3) und dem realen Szenario (vgl. Abschnitt 5.6.2) bei Anwendung der beschriebenen Logik-Ebene



(a) Basisverfahren

(b) Basisverfahren mit Warteschlangenerweiterung (Länge 1s)

(c) Basisverfahren mit sektorbezogener Warteschlangenerweiterung (9 Sektoren, 3 Bilder pro Sektor)

(d) Basisverfahren mit sektorbezogener Warteschlangenerweiterung (9 Sektoren, 6 Bilder pro Sektor)

Abbildung 5.8: Fehlerraten für verschiedene Konfigurationen des SURFtogether-Verfahrens in einem realen Szenario.

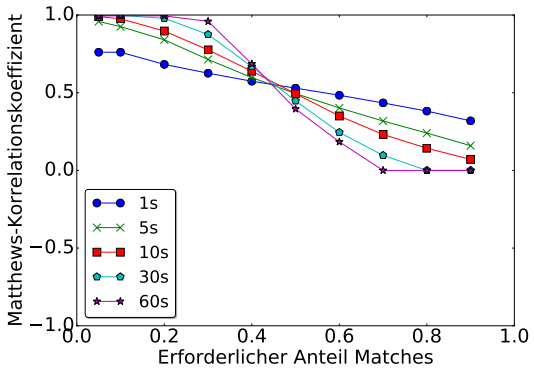
erläutert. Es wurden Zeitfenster zwischen einer und 60 Sekunden untersucht. Die Ergebnisse sind in Abbildung 5.9 dargestellt.

Im Simulations-Szenario weisen sowohl das Basisverfahren (vgl. Abbildung 5.9a) als auch das Basisverfahren mit Warteschlangenerweiterung (vgl. Abbildung 5.9b) sehr gute Werte auf. Bei Verwendung eines 30 Sekunden großen Zeitfensters und einer erforderlichen Anzahl an „Matches“ – d.h. Einzelvergleichen, die auf eine Proximität hindeuten – von 10% erreichen beide Varianten einen MCC-Wert von fast optimalen 0.99.

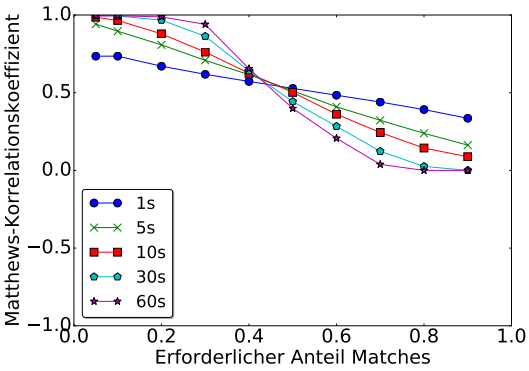
Mit der gleichen Konfiguration erreicht das mit der Logik-Ebene verbesserte Basisverfahren im realen Szenario einen MCC-Wert von 0.8 im Vergleich zu maximal 0.73 ohne Verwendung des Verfahrens. Das Basisverfahren mit Wartschlangen-Erweiterung führt in diesem Fall zu keiner Verbesserung der Ergebnisse.

5.7 Diskussion und Zusammenfassung

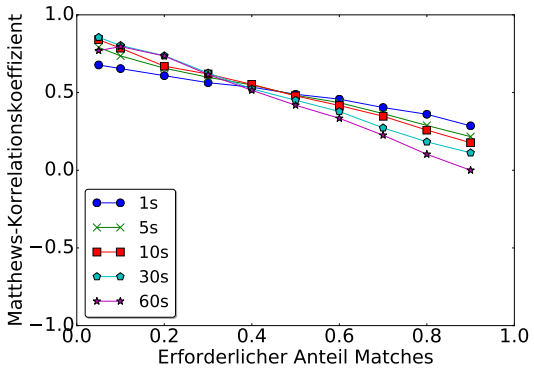
In diesem Kapitel wurde ein Verfahren zur Proximitätserkennung auf Basis visueller Daten vorgestellt und evaluiert. Die Grundidee des Verfahrens besteht darin, dass Personen, die zusammen unterwegs sind und eine Gruppe bilden, in vielen Fällen auch ähnliche Dinge sehen. Das Blickfeld der Personen kann durch am Körper getragene Kameras aufgezeichnet und analysiert werden. Aus diesen Bildaufnahmen der Umgebung – meist mit sich bewegenden Objekten bzw. Personen – kann mit Hilfe des Merkmalspunktverfahrens SURF eine abstrakte Repräsentation abgeleitet werden. Diese besitzt weder eine Ortssemantik noch gibt sie die eigentlichen Bilddaten preis. Sie kann jedoch mit anderen Repräsentationen dieser Art verglichen werden kann, um eine Proximitätsabschätzung durchzuführen.
Das Verfahren benötigt keine dedizierte Infrastruktur und keine spezielle Hardware abgesehen von Standard-Kameras wie sie in vielen aktuellen und zukünftigen Wearables verbaut sind. In verschiedenen Experimenten stellte sich heraus, dass das Verfahren nicht nur die räumliche Nähe der Testpersonen berücksichtigt, sondern implizit auch von der Übereinstimmung der Aktivitäten abhängig ist, um Aussagen bzgl. (kontexttueller) Proximität zu treffen.
Es wurde gezeigt, dass das Verfahren durch die Abhängigkeit von dynamischen Objekten im Bild robust gegenüber Fälschungsversuchen ist. Demgegenüber besteht eine gewissen Gefahr, die Privatsphäre der Benutzer zu untergraben, wenn Bild-Datenbanken mit zugehörigen Ortsinformationen zur Verfügung stehen. Diese Gefahr ließe sich ähnlich wie beim ProbeTag-Verfahren durch die Kombination mit einem PPT-Verfahren ausmerzen.



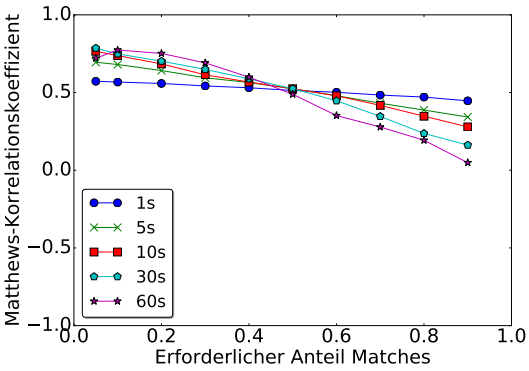
(a) Standard-Szenario der Simulation, Basisverfahren



(b) Standard-Szenario der Simulation, Basisverfahren mit Warteschlangenerweiterung (Länge 1s)



(c) Reales Szenario, Basisverfahren



(d) Reales Szenario, Basisverfahren mit Warteschlangenerweiterung (Länge 1s)

Abbildung 5.9: Matthews-Korrelationskoeffizient für verschiedene Konfigurationen der SURFtogether-Verfahrens in unterschiedlichen Szenarien. Es sind jeweils verschiedene Fenstergrößen von 1 bis 60s dargestellt.

6 Proximitätserkennung durch feingranulare Aktivitätserkennung

In den vorangegangenen Kapiteln wurden zwei verschiedene Ansätze zur Proximitätserkennung vorgestellt, die in unterschiedlichem Maße auf den Kontextelementen Ort (bzw. örtliche Nähe) und Aktivität basieren. Der WLAN-gestützte ProbeTag-Ansatz (vgl. Kapitel 4) ist dabei rein ortsbasiert, d.h. die Ähnlichkeit der erzeugten Location-Tags zweier Nutzer am gleichen Ort hängt nicht von der Aktivität der Nutzer ab.[2] Im Gegensatz dazu ist das SURFtogether-Verfahren (vgl. Kapitel 5) zusätzlich von der Benutzeraktivität abhängig, sofern sie sich auf die Blickrichtung des Benutzers auswirkt.

In diesem Kapitel wird nun ein dritter Ansatz zur Proximitätserkennung vorgestellt, der sich rein auf die Aktivität der Benutzer stützt. Die Grundidee dabei ist sowohl intuitiv nachvollziehbar als auch in der Verhaltensforschung schon seit langer Zeit bekannt: Menschen in einer Gruppe verhalten sich ähnlich [33]. Stimmen also die Aktivitäten bzw. vielmehr die Aktivitätssequenzen von Benutzern überein, so kann daraus geschlossen werden, dass sich diese in einer Gruppe, also in großer Proximität befinden.

Die folgenden Ausführungen unterscheiden sich bzgl. der Reihenfolge der Erläuterungen von den vorherigen beiden Kapiteln. Es wird in einem ersten Schritt zunächst gezeigt, dass die Aktivitätssequenz einer Person, die sich im öffentlichen Raum bewegt, nach einer bestimmten Zeit eindeutig ist, d.h. dass im Falle zweier gleicher Sequenzen, die durch eine Gruppenbildung entstanden sind, diese beiden in der Gesamtpopulation identifiziert werden können, eine Proximitätserkennung also möglich ist. Im Zuge dessen wird jedoch auch die Notwendigkeit offensichtlich werden, eine sehr feingranulare Aktivitätserkennung durchführen zu können

Eine solch feingranulare Aktivitätserkennung auf mobilen Endgeräten ist bisher noch nicht umgesetzt worden. Um die Machbarkeit zu demonstrieren, wird im zweiten Teil dieses Kapitels ein Ansatz zur feingranularen Erkennung von Aktivitäten von Personen, die sich zu Fuß oder mit der U-Bahn fortbewegen, vorgestellt.

[2] Bei ganz exakter Betrachtung kann eine physische Bewegung bzw. Positionsänderung der Endgeräte durch Einfluss auf die Empfangsstärke durchaus eine Auswirkung auf die ProbeTags haben. Diese ist jedoch im Vergleich zu den anderen Ansätzen vernachlässigbar.

6.1 Vorveröffentlichungen

Die Kerninhalte des Unterkapitels 6.3 bzgl. der feingranularen Aktivitätserkennung wurden vom Autor bereits in [125] publiziert. Wie in Kapitel 1.3 dargestellt, stammen die im Paper und im Folgenden präsentierten Inhalte bzgl. der Idee, des Konzepts und der Evaluation vom Autor. Ebenfalls im Paper bereits enthalten sind die Abbildungen 6.2 und 6.3. Die Inhalte bzgl. der Proximitätserkennung aus Aktivitätssequenzen (Abschnitt 6.2) dienten dem Paper zwar auch als Motivation, sind in diesem jedoch nicht enthalten und werden im Rahmen dieser Arbeit zum ersten Mal veröffentlicht.

6.2 Proximitätserkennung aus Aktivitätssequenzen

Der in der Verhaltenspsychologie bekannte Chamäleon-Effekt [33] besagt, dass sich Menschen, die in einer sozialen Verbindung stehen, d.h. im Prinzip eine Gruppe bilden, sich gegenseitig spiegeln, d.h. nachahmen und ähnliche körperliche Aktionen ausführen. Während der beschriebene Effekt besonders bei Mikrogesten wie der Mimik oder z.B. beim Gähnen zum Tragen kommt – und zudem auch in Wechselwirkung mit der Sympathie der Personen untereinander steht – ist auf etwas gröberer Ebene die Annahme leicht nachzuvollziehen: Personen, die sich in einer Gruppe befinden, ähneln sich auch in ihrer Aktivität. Besonders stark kommt dies, analog zum SURFtogether-Verfahren (vgl. Kapitel 5), bei der gemeinsamen Fortbewegung zu Fuß und – wie in diesem Fall zusätzlich betrachtet – mit öffentlichen Verkehrsmitteln zum Tragen. Personen, die in diesem Szenario gemeinsam unterwegs sind, betreten parallel die U-Bahn-Station, laufen oder fahren meist mit derselben Methode in das Untergeschoss, warten gleich lange auf die U-Bahn, erleben denselben Fahrtablauf und steigen auch zur gleichen Zeit wieder aus.
Je länger man diese Aktivitätssequenzen erfasst, desto eindeutiger sollten sie in einer größeren Menge von Personen werden. Lassen sich nun aber Aktivitätssequenzen von Personen eindeutig einander zuordnen, so kann daraus wiederum auf die Bildung einer Gruppe und damit die Proximität der Personen geschlossen werden.
Dies ist die Grundidee der Proximitätserkennung aus Aktivitäten. In Kapitel 3.4 wurden bereits einige verwandte Arbeiten beschrieben, die entweder durch Beobachtung von außen (meist mittels Computer-Vision-Techniken) oder durch Aufzeichnung diverser Sensordaten an den Subjekten und Vergleich dieser versuchen, auf Gruppenkonstellationen zu schließen. Methoden, die wie die erstgenannten meist Computer-Vision-gestützten Verfahren eine dedizierte Infrastruktur benötigen und damit nur in vorher vorbereiteten Bereichen funktionieren, sind im Hinblick auf die gestellten Anforderungen (vgl. Kapitel 2.4.5) keine geeigneten Kandidaten für die betrachteten Einsatzzwecke.

Die sensor-basierten, am Subjekt aufzeichnenden Verfahren sind dagegen vielversprechend. Eine Frage, die bisher nicht betrachtet wurde, ist, inwiefern sich die Gruppenkonstellationen auch bei sehr großen Nutzerzahlen eindeutig erkennen lassen. Die betrachteten Experimente beziehen sich meist nur auf mehrere zehn bis hundert Testsubjekte, die im Verlauf des Experiments Gruppen formen und wieder auflösen. Ob sich mit diesen Verfahren Gruppen auch eindeutig bei mehreren zehntausend Benutzern identifizieren lassen könnten, bleibt offen.
Weiterhin verwenden die vorgestellten Verfahren eine sehr große Menge an Sensordaten beim Vergleich und Test auf Gruppensituationen, d.h. die gesamte Datenmenge muss zu der Instanz transferiert werden, die die Proximitätsbestimmung vornimmt. Dies ist im Falle des in Kapitel 2.1 skizzierten Systemmodells der PEC, der sich entweder auf einer zentralen Server-Instanz oder bei jedem beteiligten Benutzer individuell befindet. Die großen Datenmengen müssen also über das Netzwerk transferiert werden, was für mobile Endgeräte aus Bandbreiten, Kosten und Energie-Sicht ein Problem darstellt.

Es besteht also die Notwendigkeit, die zum Vergleich benötigten Sensordaten bereits am Endgerät zu aggregieren, z.B. indem dort bereits höherwertige Aktivitäten aus den rohen Daten bestimmt und dann nur diese Informationen an die betroffenen Komponenten versandt werden. Dadurch gehen jedoch feinere Unterscheidungsmöglichkeiten verloren, sodass es wiederum einfacher wird, Proximitätssituationen falsch einzuschätzen, weil sich die so gebildeten Aktivitäten zufällig gleichen (oder die übermittelten Informationen absichtlich so gewählt wurden, um eine Proximitätssituation zu fälschen).
Als Mittelweg bietet es sich an, Aktivitätssequenzen auf den Endgeräten zu aggregieren, und dann diese Sequenzen untereinander auszutauschen. Intuitiv ist zu vermuten, dass je länger der Zeitraum der betrachteten Sequenz ist, desto eindeutiger sich diese auch von anderen unterscheiden lässt. Die Aktivitätssequenzen können analog zu den bisher betrachteten Location Tags sogar als eine Art Aktivitäts-Tag oder, noch allgemeiner, als Kontext-Tag aufgefasst werden.
Es stellt sich die Frage, welche Aktivitäten bei der Bildung solcher Sequenzen verwendet werden sollen. Ebenso wie die Länge der Sequenz spielt auch die zeitliche Dauer der einzelnen Aktivitäten eine Rolle. Betrachtet man eine Sequenz der Länge 120 Sekunden und dauern die verwendeten Aktivitäten im Schnitt 60 Sekunden an, so wird man in der Sequenz nicht viele Aktivitätswechsel, und damit Unterscheidungsmerkmale zu anderen Sequenzen finden können. Im Gegensatz dazu würden Aktivitäten mit nur kurzen Zeitintervallen von nur wenigen Sekunden zu einer deutlich größeren Variabilität in der Sequenz führen.
Um kurz dauernde Aktivitäten als Grundlage für die Aktivitätssequenzen verwenden zu können, ist eine sehr feingranulare Aktivitätserkennung notwendig. Die aktuell in den mobilen Betriebssystemen enthaltenen Aktivitätser-

kennungsdienste beschränken sich bei den (mehr oder weniger) erkennbaren Aktivitäten auf grundsätzliche Aspekte wie „stehen“, „gehen“, „rennen“, „mit dem Fahrrad fahren“ und „mit dem Auto fahren“ [7, 69]. Für das in Kapitel 2.4.5 erläuterte primäre Szenario, dass sich Benutzer zusammen zu Fuß fortbewegen, sind aus dieser Menge nur zwei Aktivitäten („stehen“ und „gehen“) tatsächlich interessant. Selbst „rennen“, das in den Betriebssystemen primär zur Aufzeichnung von sportlicher Aktivität verwendet wird, ist in Alltagssituationen nicht von Bedeutung.
Demgegenüber gibt es unter den verwandten Forschungsarbeiten zur Aktivitätserkennung eine Vielzahl von Ansätzen, die auch andere Aktivitäten erkennen können. Ein Überblick darüber wird später in Kapitel 6.3.2 noch gegeben. Zunächst soll nun aber erläutert werden, inwiefern sich die Feingranularität der Aktivitätserkennung tatsächlich auf die eindeutige Identifizierbarkeit einzelner Benutzer bzw. Gruppen auswirkt.

6.2.1 Betrachtetes Szenario für die Proximitätserkennung aus Aktivitätssequenzen

Um eine Abschätzung des Mehrwertes von feingranularer Aktivitätserkennung gegenüber den aktuell bereits in den Betriebssystemen verfügbaren Möglichkeiten zur Aktivitätserkennung treffen zu können, wird im Folgenden eine spezielle Form des in Kapitel 2.4.5 definierten primären Anwendungsszenarios der Proximitätserkennung verwendet. Es handelt sich dabei um die Fortbewegung von Personen im öffentlich Nahverkehrsnetz, genauer gesagt in einem U-Bahn-Netz.
In diesem Szenario sind die Personen allgemein gesehen „zu Fuß“ unterwegs, da sich ihre eigenen körperlichen Aktivitäten im Wesentlichen aus „stehen“ und „gehen“ zusammensetzen, egal ob in der U-Bahn-Station oder in einem U-Bahn-Wagon. Es handelt sich also um die beiden Aktivitäten, die mit den bereits auf mobilen Endgeräten allgemein verfügbaren Mitteln erkannt werden können.
Desweiteren wurde eine Liste vor Aktivitäten definiert, welche die beiden vorher genannten Aktivitäten feiner untergliedern und im Allgemeinen bei der Fortbewegung im U-Bahn-Netz auftreten können. Die Aktivitäten sind in Tabelle 6.1 zusammengefasst. Ein typischer Ablauf bei der Benutzung der U-Bahn sieht in der Regel so aus, dass Person zunächst zu Fuß zur U-Bahn-Station geht und diese betritt (*walk*). Da sich die Gleise meist tiefer unter der Erde befinden, bewegt sich die Person entweder per Treppe (*walkdown*), per Rolltreppe (*rolldown*, *rolldownwalk*) oder per Fahrstuhl (*drivedown*) auf die untere Ebene, wobei auf einer Rolltreppe alternativ gestanden (*rolldown*) oder gelaufen (*rolldownwalk*) werden kann. Auf der unteren Ebene bewegt sich die Person zum Gleis (*walk*) und muss dort auf die U-Bahn warten (*walkwait*). Nach dem Eintreffen der U-Bahn (*subarrive*) kann die Person in diese Einsteigen (*subenter*) und wartet darin (*subwait*) bis dies losfährt (*subaccel*). Die U-Bahn fährt

Bezeichnung	Bedeutung	Hauptklasse
walk	Normales Gehen	gehen
walkwait	Stehen (außerhalb der U-Bahn)	stehen
walkup	Treppe hochgehen	gehen
walkdown	Treppe hinunter gehen	gehen
rollupwalk	Eine fahrende Rolltreppe nach oben gehen	gehen
rolldownwalk	Eine fahrende Rolltreppe nach unten gehen	gehen
rollup	Eine Rolltreppe nach oben fahren	stehen
rolldown	Eine Rolltreppe nach unten fahren	stehen
driveup	Mit einem Fahrstuhl nach oben fahren	stehen
drivedown	Mit einem Fahrstuhl nach unten fahren	stehen
subarrive	Warten während eine U-Bahn einfährt	stehen
subenter	Einsteigen in die U-Bahn	gehen
subwait	Stehen in der U-Bahn während diese steht	stehen
subaccel	Stehen in der U-Bahn während diese losfährt	stehen
subdrive	Stehen in der U-Bahn während diese fährt	stehen
subbrake	Stehen in der U-Bahn während diese abbremst	stehen
subexit	Aussteigen aus der U-Bahn	gehen

Tabelle 6.1: Betrachtete Aktivitäten im U-Bahn-Szenario. Die Hauptklasse stellt die Einordnung der Aktivität in eine der primären Klassen „gehen“ oder „stehen“ dar.

dann einige Zeit (*subdrive*) bis sie wieder abbremst (*subbrake*) und die Person am Ende aussteigen kann (*subexit*). Fährt die Person mehrere Stationen, so wiederholen sich U-Bahn-bezogenenen Aktivitäten mehrfach bevor die Person aussteigt. Am Ende der fahrt benutzt die Person erneut entweder die Treppe (*walkup*), die Rolltreppe (*rollup*, *rollupwalk*) oder den Fahrstuhl (*driveup*), um wieder die Oberfläche zu erreichen.
Insgesamt lassen sich so also 17 verschiedene Aktivitäten unterscheiden. Im Folgenden wird nun gezeigt, inwieweit sich eine in dieser Form feingranularere Aktivitätserkennung auf die Eindeutigkeit von Aktivitätssequenzen auswirkt.

6.2.2 Simulation der Fortbewegung in einem U-Bahn-Szenario

Um die Effekte unterschiedlich feingranularer Akitvitätserkennungsmethoden auf die eindeutige Identifizierbarkeit einer Aktivitätssequenz bei einer großen Gesamtmenge von Nutzern – und damit die Eignung zur Proximitätserkennung – zu untersuchen, wurde eine Simulation implementiert. Dazu wurde das U-Bahn-Netz der Stadt München nachmodelliert mit allen U-Bahn-Stationen und U-Bahn-Linien. Die U-Bahnen verkehren in der Simulation entlang der

realen Strecke, wobei jedoch nicht der echte Fahrplan zugrunde gelegt wurde, sondern ein fester Takt von drei Minuten. Die Zeitdauer der unterschiedlichen Bewegungsphasen der U-Bahnen (stehen, anfahren, fahren, etc.), die den jeweiligen Aktivitäten aus Tabelle 6.1 zuzuordnen sind, wurde jeweils mit dem Durchschnittswert belegt, der bei einer realen Datenaufzeichnung, die später in Kapitel 6.4.1 noch genauer erläutert wird, beobachtet wurde.

In diesem so modellierten U-Bahn-Netz werden 50.000 Fahrgäste simuliert, die sich zeitgleich darin fortbewegen. Diese Zahl entspricht ungefähr dem durchschnittlichen Fahrgastaufkommen des Münchner U-Bahn-Netzes [142]. Zu Stoßzeiten kann diese Zahl auch deutlich höher liegen. Die Größenordnung ist im Verhältnis zur Größe des U-Bahn-Netzes jedoch passend, was für die Simulation ausreichend ist.

Jeder der simulierten Fahrgäste bewegt sich individuell durch das U-Bahn-System. Dazu wird der Fahrgast zu Beginn an einer zufällig gewählten U-Bahn-Station instanziiert und eine weitere zufällige U-Bahn-Station als Ziel ausgewählt. Auf dem U-Bahn-Graphen wird anschließend der kürzeste Pfad dorthin berechnet. Der Fahrgast begibt sich nun in die U-Bahn-Station, geht bzw. fährt in das Untergeschoss und wartet dort auf die passende U-Bahn. Ist die richtige Linie mit der richtigen Endstation, d.h. der richtigen Fahrtrichtung, angekommen, so steigt der Fahrgast ein. Da die U-Bahn-Züge tatsächlich simuliert werden, können sich also auch mehrere Fahrgäste an der Station ansammeln, die dann alle gleichzeitig in die U-Bahn einsteigen. Solange sich der Fahrgast in der U-Bahn aufhält, wird seine aktuelle Aktivität anhand der entsprechenden Bewegungsphase der U-Bahn bestimmt, d.h. insbesondere auch, dass hier die Aktivitäten aller Fahrgäste in der selben U-Bahn synchron und identisch sind.

Hat der Fahrgast die Zielstation erreicht, so steigt er aus und verlässt die U-Bahn-Station über eine der bereits beschriebenen Möglichkeiten. Die Fahrgastinstanz wird nun entfernt und gleichzeitig eine neue erstellt, die wiederum von einer zufällig gewählten U-Bahn-Station aus startet.

Für den Fall, dass eine Route es erfordert, zwischendurch umzusteigen, so wird auch dies simuliert. Der Fahrgast steigt dazu an der entsprechenden Station aus und durchläuft die gleichen Phasen wie beim Beginn der Fahrt, um schließlich in die U-Bahn der Anschlusslinie einzusteigen.

Welche Methode ein Fahrgast wählt, um in die U-Bahn-Station hinunter bzw. zurück zur Oberfläche zu gelangen, wird aus den vier Möglichkeiten Treppe, Rolltreppe (stehend), Rolltreppe (gehend) und Aufzug zufällig ausgewählt. Die Zeitdauern dieser Aktivitäten werden zufällig aus dem jeweiligen Intervall, das bei einer echten Datenaufzeichnung (vgl. später Kapitel 6.4.1) beobachtet wurde, ausgewählt. Die Dauer von Aktivitäten, die in Bezug zu den U-Bahn-Zügen stehen, werden durch die tatsächlich simulierten Fahrbewegungen der Züge bestimmt.

6.2.3 Evaluation der Eignung von Aktivitätssequenzen zur Proximitätserkennung

Mit Hilfe der entwickelten Simulation wurde ein Zeitraum von 24 Stunden mit durchgehend 50.000 Fahrgästen simuliert. Für jeden dieser Fahrgäste wurde die Abfolge der in Tabelle 6.1 aufgeführten Aktivitäten aufgezeichnet.

Aus diesem Datensatz wurden für 20 verschiedene Intervallgrößen zwischen einer und 7200 Sekunden jeweils 200 Fahrgäste zufällig aus der Gesamtmenge ausgewählt. Für jeden dieser Fahrgäste wurde für einen zufälligen Zeitpunkt ein Fenster der entsprechenden Größe aus dem Datensatz gewählt und die in diesem Zeitraum durchlaufene Aktivitässequenz extrahiert. Dabei wurde rein die Reihenfolge unterschiedlicher Aktivitäten betrachtet, d.h. die individuellen Längen der Aktivitäten spielen keine Rolle (nur implizit, da bei länger andauernden Aktivitäten insgesamt weniger Aktivitätswechsel innerhalb des betrachteten Zeitfensters stattfinden).

Diese so gewonnene Aktivitätssequenz wurde dann mit den auf die gleiche Weise extrahierten Sequenzen aller anderen 49.999 Fahrgäste zum gleichen Zeitpunkt verglichen. So konnte für jede Zeitfenstergröße eine Abschätzung der Anzahl identischer Aktivitätssequenzen berechnet werden.

Zum Vergleich wurde für jede betrachtete Aktivitätssequenz die gleiche Berechnung durchgeführt, nachdem die Sequenz auf die beiden Basisaktivitäten „gehen“ und „stehen“ vereinfacht wurde. Das heißt, jede Aktivität der Sequenz wurde anhand der in Tabelle 6.1 aufgeführten Zuordnung auf eine dieser beiden Hauptklassen abgebildet, sodass die Sequenz nur noch aus eben diesen beiden Aktivitäten im Wechsel bestand.

Die Ergebnisse der feingranularen Aktivitätserkennung sind im Vergleich zur einfachen Aktivitätserkennung in Abbildung 6.1 dargestellt. Horizontal sind hierbei die veschiedenen Längen der extrahierten Aktivitätssequenzen dargestellt, wobei für jede Fenstergröße jeweils das Ergebnis der feingranularen (blau) und der einfachen Aktivitätserkennung (grün) dargestellt ist. Auf der Ordinate ist der Anteil der gefundenen identischen Aktivitätssequenzen an der Gesamtmenge dargestellt. Je höher der Wert, mit desto mehr anderen Fahrgästen stimmt die Aktivitätssequenz der zufällig ausgewählten Fahrgäste also überein. Gewollt ist hier eine möglichst niedrige Anzahl.

Zur Darstellung werden Boxplots verwendet, die für jede Messreihe das arithmetische Mittel, den Median, das untere und obere Quartil, sowie das 2%-Quantil und das 98%-Quantil enthalten. Ähnlichkeitswerte, die nicht in den 2%-98%-Quantilsbereich fallen, werden einzeln im Plot dargestellt.

Es ist deutlich zu erkennen, dass die feingranulare Aktivitätserkennung bereits bei sehr kurzen Fenstern eine deutlich bessere Unterscheidung der Sequenzen ermöglicht als die alleinige Verwendung der zwei Aktivitäten „gehen“ und „stehen“. Bei Sequenzlängen ab 90 Sekunden liegt der Median des Anteils der identischen Sequenzen bereits bei unter 10%, während er bei der einfachen Aktivitätserkennung noch über 80% beträgt. Ab einer Sequenzdauer von

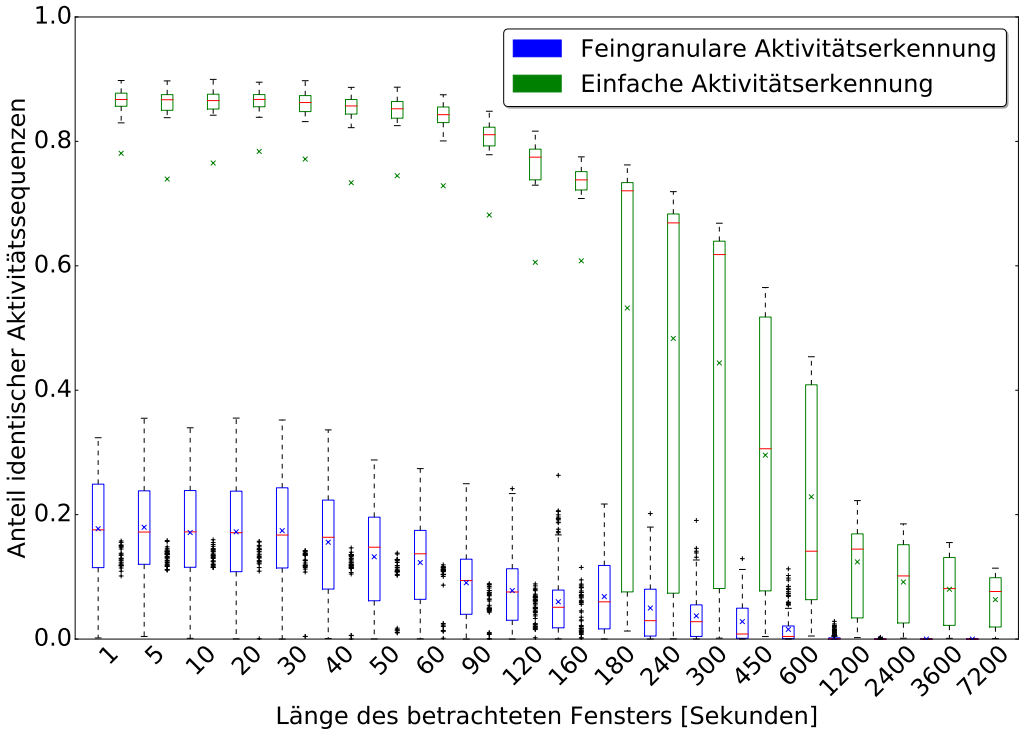


Abbildung 6.1: Vergleich der Eindeutigkeit von Aktivitätssequenzen unterschiedlicher Länge bei Verwendung von feingranularer Aktivitätserkennung gegenüber einfacher Aktivitätserkennung.

siebeneinhalb Minuten stimmen mit der feingranularen Aktivitätserkennung sogar nur noch ca. 1% der Sequenzen überein, während nur mit Hilfe der Basisaktivitäten immer noch rund 30% Übereinstimmungen gefunden werden. Allgemein ist im Hinblick auf die Zeitfenstergrößen zu erwähnen, dass diese gegenüber einer intuitiven Einschätzung erstaunlich lang erscheinen, wenn eine niedrige Übereinstimmungsrate betrachtet wird. Dies ist auf die Verwendung einer Simulation zurückzuführen, die die Vielfalt der echten Welt natürlich nur bedingt abbilden kann. Die absoluten Werte der Simulationsergebnisse sind also nicht 1-zu-1 auf die reale Welt zu übertragen, insbesondere die benötigten Zeitfenstergrößen, die in der Realität deutlich kürzer ausfüllen dürften. Als relative Einordnung der feingranularen gegenüber der einfachen Aktivitätserkennung sind die Simulationsergebnisse jedoch hilfreich.

Zusammenfassend lässt sich festhalten, dass eine Proximitätserkennung auf Basis von Aktivitätssequenzen möglich erscheint. Mit zunehmender Länge der betrachteten Aktivitätssequenzen und insbesondere auch mit der Granularität, mit der die Aktivitäten erkannt werden können, werden die Aktivitätssequenzen immer eindeutiger in der Gesamtmenge identifizierbar. Damit ist im Falle einer Proximitätssituation, d.h. z.B. einer Gruppensituation, in der sich die Mitglieder zusammen im U-Bahn-Netz fortbewegen, und der damit innerhalb

der Gruppe übereinstimmenden Aktivitätssequenz, eine Unterscheidung der Gruppe von der restlichen Population möglich und damit die Proximität erkennbar.

Es ist einschränkend anzumerken, dass der Vergleich auf Identität der Sequenzen auch innerhalb einer Gruppe mit zunehmender Feingranularität der Aktivitäten in der Praxis häufiger fehlschlagen wird, da sich die Mitglieder zwar ähnlich, aber nicht komplett gleich verhalten. Zudem muss grundsätzlich auch bei den Aktivitätserkennungsverfahren davon ausgegangen werden, dass diese nicht komplett fehlerfrei arbeiten. Für eine tatsächliche Umsetzung sollten die Aktivitätssequenzen also nicht auf Gleichheit sondern auf den Grad der Ähnlichkeit getestet werden, was u.a. mit Hilfe von Methoden zum String-Vergleich aus der Textanalyse bzw. der Bioinformatik umsetzbar wäre [145].

Für die weiteren Ausführungen ist jedoch ausreichend gezeigt zu haben, dass eine feingranulare Aktivitätserkennung die Umsetzung von Proximitätserkennung auf Basis von Aktivitäten in erheblichem Maße erleichtern bzw. effizienter machen würde. Deswegen wird im folgenden Teil die Machbarkeit einer ebensolchen feingranularen Aktivitätserkennung im hier betrachteten U-Bahn-Szenario gezeigt.

6.3 Feingranulare Aktivitätserkennung bei der Fortbewegung mit der U-Bahn

Die Aktivität eines Benutzers wird im Allgemeinen als primäres Kontextelement betrachtet (vgl. dazu Kapitel 2.2.2). Eine Erkennung dieser ermöglicht im Sinne des Konzepts der kontextbezogenen Anwendungen eine Anpassung der Diensterbringung an die aktuelle Situation des Benutzers. Je vielfältiger und feiner Aktivitäten erkannt werden, desto mehr Möglichkeiten ergeben sich zur sinnvollen Reaktion darauf.

Im Rahmen dieser Arbeit liegt der hauptsächliche Nutzen des Einbezugs von Aktivitäten in die Diensterbringung in der damit möglichen Verbesserung bzw. Ermöglichung der angestrebten Proximitätserkennung. Personen, die eine Gruppe bilden, verhalten sich ähnlich und können dadurch einander zugeordnet werden. Dies kann implizit eine Rolle spielen, wie beim vorgestellten SURFtogether-Verfahren (vgl. Kapitel 5) oder explizit wie beim im vorherigen Abschnitt skizzierten Ansatz (vgl. Abschnitt 6.2). Trotz dieser speziellen Motivation zur Entwicklung von Möglichkeiten zur feingranularen Aktivitätserkennung sind die im Folgenden erläuterten Methoden und Erkenntnisse auch für viele andere Anwendungsbereiche relevant.

Das Ziel der Aktivitätserkennung ist in diesem Fall die Erkennung und Unterscheidung der 17 verschiedenen in Tabelle 6.1 aufgeführten Aktivitäten, die bei der Fortbewegung einer Person zu Fuß und insbesondere mit der U-Bahn auftreten können.

6.3.1 Anforderungen an das Verfahren zur Aktivitätserkennung

Die Vielzahl von Ansätzen im Bereich der Aktivitätserkennung unterscheidet sich nicht nur in den zu erkennenden Aktivitäten und den verwendeten Verfahren, sondern auch in den generellen Rahmenbedingungen, die im Voraus definiert werden. Daher ist es notwendig, bestimmte Anforderungen festzuhalten, die vom entwickelten Erkennungsverfahren erfüllt werden müssen bzw. innerhalb dieser Rahmenbedingungen es einsetzbar sein muss.
Da im vorliegenden Anwendungsfall die Aktivitätserkennung nicht der Erkennung der Aktivität an sich dient, sondern der damit möglichen Proximitätserkennung, muss das Verfahren für den Benutzer so unaufwendig und unauffällig wie möglich sein. Während z.B. beim Tracking von sportlichen Aktivitäten durchaus dedizierte Geräte zum Einsatz kommen können, da die Aktivitätserkennung einen direkten Mehrwert bei der Durchführung der Aktivität bietet, sind die in Kapitel 2.3 gezeigten Anwendungsfälle in ihren Auswirkungen weniger direkt.
Ein angemessenes Verfahren muss daher insbesondere die folgenden beiden Rahmenbedingungen einhalten:

- Das Verfahren muss alleine unter Nutzung eines modernen Smartphones mit üblicher Hardware und Sensor-Ausstattung umsetzbar sein, d.h. insbesondere, dass die Person keine zusätzliche Sensorik z.B. an den Gliedmaßen tragen muss.

- Das Verfahren muss unabhängig von der Trageposition des Endgerätes funktionieren, d.h. insbesondere, dass der Benutzer nicht gezwungen sein darf, das Endgerät aktiv in der Hand zu halten

Damit ist sichergestellt, dass das Verfahren unaufdringlich im Hintergrund arbeiten kann, ohne den Benutzer zu beeinträchtigen.

6.3.2 Verwandte Arbeiten im Bereich der Aktivitätserkennung

Im Folgenden wird ein Überblick über verwandte Arbeiten aus dem Bereich der Aktivitätserkennung gegeben. Nach einer Betrachtung ausgewählter allgemeiner Ansätze aus diesem Themenbereich werden auch noch explizit Verfahren zur Erkennung der Fortbewegungsart eines Nutzers vorgestellt.

6.3.2.1 Erkennung von Aktivitäten

Die ersten Arbeiten zur Erkennung von Aktivitäten eines Benutzers mit Hilfe von Sensordaten verwendeten (meist mehrere) dedizierte, am Körper angebrachte Beschleunigungssensoren. Bao et al. [16] statten ihre Testpersonen mit fünf Beschleunigungssensoren aus, getragen am Handgelenk, Ellenbogen,

Knie, Schienbein und an der Hüfte. Unter Verwendung verschiedener Algorithmen des maschinellen Lernens versuchen sie 20 verschiedene mehr oder weniger alltägliche Aktivitäten zu erkennen. Darunter befinden sich Aktivitäten wie Gehen, Sitzen, Fernsehen, Wäsche zusammenlegen, Zähne putzen oder Staubsaugen. Sie erreichen dabei eine durchschnittliche Erkennungsrate von 84%.

Ravi et al. [154] verwenden nur einen an der Hüfte getragenen Beschleunigungssensor und beschränken sich zudem auf acht Aktivitäten. Letztere stammen erneut aus dem Bereich der Alltagsaktivitäten wie Gehen, Stehen, Staub saugen oder Zähneputzen. Das Hauptaugenmerk der Arbeit liegt jedoch weniger auf der Erkennung der Aktivitäten an sich, sondern vielmehr auf der Einschätzung der Tauglichkeit von Ensemble-Methoden wie Bagging [29] und Boosting [58] zur Verbesserung der Klassifikationsergebnisse. Das Ergebnis fällt positiv zugunsten der Ensemble-Methoden aus. Dies ist im Einklang mit den in der vorliegenden Arbeit beobachteten Ergebnissen, die beim Random Forest Algorithmus als Ensemble-Methode am besten ausfallen (vgl. Kapitel 6.4.2).

Mit dem Aufkommen leistungsfähiger Smartphones mit umfangreicher integrierter Sensorik fokussierte sich die Forschung im Bereich der Aktivitätserkennung mehr und mehr auf diese Endgeräte. Eine Herausforderung stellt dabei die nun nicht mehr festgelegte Trageposition des Sensors dar, da sich das Smartphone z.B. in der Hand, im Rucksack oder in der Hosentasche des Trägers befinden und die Position auch wechseln kann. Sun et al. [176] schlagen unter anderem vor, die Aktivitätserkennung in einem zweistufigen Verfahren durchzuführen. Zunächst wird die Trageposition des Endgerätes an sich erkannt und erst im zweiten Schritt wird mit einem dedizierten Klassifikator für genau diese Position die Aktivität bestimmt. Es wird erneut versucht, typische alltägliche Aktivitäten wie Gehen und Stehen zu erkennen, was den Autoren auch mit über 94% Genauigkeit gelingt.

Alternativ kann die unbekannte Trageposition des Endgeräts auch durch die Verwendung positionsunabhängiger Features kompensiert werden. Dies versuchen Khan et al. [99] indem sie mit Hilfe einer *Kernel Discriminant Analysis* die Features identifizieren, deren Werte sich innerhalb einer Aktivitätsklasse unabhängig von der Trageposition nur geringfügig unterscheiden und deren Werte gleichzeitig deutlich verschieden von denen der anderen Aktivitäten sind. Für fünf typische Aktivitäten wie Gehen und Stehen erreichen sie eine Erkennungsgenauigkeit von 96%.

Während für alltägliche Aktivitäten primär Beschleunigungs- und Drehratensensoren zum Einsatz kommen, wird insbesondere für speziellere Aktivitäten oft auch auf andere Sensoren bzw. Datengrundlagen zurückgegriffen. Yatani et al. stellen mit *BodyScope* [199] einen tragbaren Akustiksensor vor, der am Hals angebracht wird und mit dem sich Aktivitäten wie Essen, Trinken, Sprechen, Flüstern oder Lachen unterscheiden lassen. Die Autoren greifen dabei auf charakteristische Muster zurück, die im Audio-Spektrogramm der verschiedenen Aktivitäten enthalten sind. Sie erreichen eine Erkennungsgenauigkeit von über

71%.
Nam et al. [143] verwenden neben Beschleunigungssensoren zusätzlich eine am Körper getragene Kamera. Aus dem Video-Stream der Kamera werden Features des *optischen Flusses* berechnet und zur Klassifikation verwendet. Die Aktivitätserkennung erfolgt durch die Kombination zweier zunächst unabhängiger Klassifizierungen, einmal anhand der Beschleunigungsdaten und einmal anhand der Video-Daten. Es wurden erneut alltägliche Aktivitäten wie Gehen und Stehen untersucht mit einer Genauigkeit von über 92% erkannt.

Bewertung Die gezeigte Auswahl an Verfahren zur Aktivitätserkennung steht repräsentativ für die aktuelle Landschaft an Ansätzen in diesem Bereich. Die gestellten allgemeinen Anforderungen bzgl. Sensorik und Unabhängigkeit der Trageposition (vgl. Abschnitt 6.3.1) sind in aktuellen verwandten Arbeiten meist erfüllt. Die jeweiligen Verfahren beziehen sich jedoch auf eine oft deutlich andere Menge an zu erkennenden Aktivitäten, sodass die dort erzielten Ergebnisse nur bedingt übertragbar sind. Alltägliche Aktivitäten wie Staubsaugen sind nicht hilfreich, um Gruppensituationen zu erkennen, da es sich i.d.R. um Individualaktivitäten handelt. Zudem sind die meisten betrachteten Aktivitäten relativ lang andauernd, sodass die gewünschte Häufigkeit an Veränderungen in einer Aktivitätssquenz nur schwer zu erreichen ist.
Andererseits können einige Ansätze für die gewünschte feingranulare Aktivitätserkennung im U-Bahn-Szenario adaptiert werden. Gerade die von Yatani et al. gezeigte Verwendung von Audio-Daten [199] hat sich auch im vorliegenden Szenario als sehr hilfreich erwiesen (vgl. Abschnitt 6.3.3). Eine Verwendung von visuellen Daten wird im Folgenden vorerst nicht weiterverfolgt. Der von Nam et al. gezeigte Ansatz [143] dient jedoch im Rückblick auf das in Kapitel 5 vorgestellte Proximitätserkennungsverfahren als Beispiel, wie auch Video-Daten auf unterschiedliche Arten zur Erkennung von Kontextelementen verwendet werden können.

6.3.2.2 Erkennung der Fortbewegungsart

Die bisher vorgestellten Ansätze befassen sich allgemein mit Aktivitätserkennung bzw. primär mit alltäglichen Aktivitäten wie Gehen und Stehen. Es existieren darüberhinaus auch einige Arbeiten, die sich dediziert mit der Erkennung der Fortbewegungsart einer Person befassen und damit ein ähnliches Szenario betrachten wie das in dieser Arbeit untersuchte U-Bahn-Szenario.
Reddy et al. [155] verwenden die GPS- und Beschleunigungssensordaten des Smartphones eines Benutzers um herauszufinden, ob der Nutzer steht, geht, rennt, Fahrrad fährt oder ein motorisiertes Fahrzeug benutzt, wobei bei letzterem nicht zwischen Auto, Motorrad, Bus, etc. unterschieden wird. Sie erreichen eine Erkennungsrate von über 90%. Allerdings erscheinen die gewählten Aktivitäten generell sehr leicht unterscheidbar: Die Aktivitäten Stehen, Gehen und Rennen sind in anderen Arbeiten vielfach betrachtet worden, und die Aktivitäten Fahrrad fahren und motorisiertes Fahren sind relativ leicht anhand

der Geschwindigkeit – die aus den GPS-Daten extrahiert werden kann – zu unterscheiden. Sehr ähnlich verhält sich im Hinblick auf die Unterscheidung der genannten Aktivitäten das *CenceMe* Framework von Miluzzo et al. [136]. Der Fokus dieser Arbeit liegt jedoch auf einem umfangreichen Framework zur Verarbeitung, Analyse und sogar Veröffentlichung von Aktivitätsdaten.
Interessanter erscheint die Arbeit von Zhang et al. [202], in der eine etwas feinere Unterscheidung der Fortbewegungsart versucht wird. Die betrachteten Aktivitäten umfassen Gehen, Fahrrad fahren, Busfahren, Autofahren als Fahrer und Autofahren als Mitfahrer. Neben den GPS- und Beschleunigungssensordaten eines Smartphones verwenden die Autoren an der Fußsohle angebrachte Drucksensoren. Gerade letztere tragen entscheidend dazu bei, Aktivitäten wie Autofahren als Fahrer und Autofahren als Mitfahrer zu unterscheiden. Insgesamt wird mit dem Verfahren eine durchschnittliche Erkennungsgenauigkeit von 95% erreicht.
Zheng et al. [204] beschränken sich wiederum rein auf GPS-Daten um die Fortbewegungsarten Gehen, Fahrrad fahren, Autofahren und Busfahren zu unterscheiden. Sie verwenden jedoch komplexere Features auf den Daten wie beispielsweise die Häufigkeit von Änderungen der Richtung, von Anhaltevorgängen oder von Änderungen der Geschwindkeit. Das Verfahren erreicht immerhin eine Klassifikationsgenauigkeit von 76%, ohne wie vergleichbare Verfahren zusätzlich auf Beschleunigungssensordaten o.ä. zurückzugreifen.
Während die bisher gezeigten Arbeiten eine (meist generische) motorisierte Fortbewegung (primär) anhand der Fortbewegungsgeschwindigkeit und ähnlicher aus den GPS-Daten errechneter Features erkennen, gehen Stenneth et al. [173] einen Schritt weiter und versuchen zusätzlich verschiedene Typen der motorisierten Fortbewegung – nämlich Busfahren, Autofahren und Zugfahren – zu unterscheiden. Hierzu verwenden sie Informationen über das Verkehrsnetz wie Bushaltestellen und Streckenverläufe der Züge. Durch Vergleich der GPS-Trajektorien der Nutzer mit dem Verkehrsnetz sowie einer generellen Aktivitätserkennung ähnlich zu den vorher beschriebenen Verfahren erreichen sie eine Erkennungsgenauigkeit von über 93%.

Bewertung Die vorgestellten Verfahren zur Erkennung der Fortbewegungsart erreichen bisher nicht den Detailgrad, der im Hinblick auf das betrachtete U-Bahn-Szenario und die gewünschte feingranulare Aktivitätserkennung notwendig ist. Ansätze, die dedizierte Sensoren – wie z.B. die Drucksensoren an den Fußsohlen beim Ansatz von Zhang et al. [202] – verwenden, scheiden zudem durch die gestellten Anforderungen aus. Darüberhinaus weisen die gezeigten Verfahren zur Erkennung der Fortbewegungsart eine weitere gemeinsame Schwäche bzgl. des U-Bahn-Szenarios auf: Die Verwendung von GPS ist unter der Erdoberfläche nicht möglich.

Es existieren also wie in diesem Abschnitt beschrieben viele verschiedene Ansätze, die für unterschiedlichste Anwendungsfälle verschiedene Aktivitäten mit

oft sehr hoher Genauigkeit erkennen bzw. unterscheiden können. Die bisherigen Arbeiten betrachten jedoch keine feingranularen Abstufungen der Aktivitäten. Analog zum im Folgenden betrachteten U-Bahn-Szenario könnte man mit den vorgestellten Verfahren beispielsweise bei Betrachtung der Fortbewegung mit normalen Zügen eine Unterscheidung zwischen der Fortbewegung zu Fuß und der Fortbewegung per Zug vornehmen. Eine deutlich feinere Abstufung, wie z.B. Einsteigen, Beschleunigen, Bremsen und Aussteigen, wurde bisher nicht gezeigt. Da letzteres wie in Abschnitt 6.2 beschrieben für eine Proximitätserkennung anhand von Aktivitätssequenzen notwendig erscheint, wird im weiteren Verlauf der Arbeit die Machbarkeit einer solch feingranularen Aktivitätserkennung demonstriert.

6.3.3 Konzept zur feingranularen Aktivitätserkennung im U-Bahn-Szenario

Im Folgenden wird das Konzept zur feingranularen Erkennung der 17 veschiedenen Aktivitäten, die im betrachteten U-Bahn-Szenario auftreten können, erläutert. Das grundsätzliche Vorgehen entspricht dem aktuellen Stand der Wissenschaft. Ein Hauptaugenmerk bei der Aktivitätserkennung liegt in der Regel auf der Auswahl geeigneter *Features*, die auf den Daten berechnet werden. Daher beschäftigen sich die folgenden Ausführungen auch hauptsächlich mit dieser Thematik.
Zur Erkennung der unterschiedlichen Aktivitäten zeichnet das Smartphone des Benutzers kontinuierlich die Sensordaten des Beschleunigungssensors, des Barometers und des Mikrofons auf. Es werden also die aus drei Komponenten (Achsen) bestehenden Werte für die Beschleunigung des Endgeräts, der Luftdruck und eine Audio-Aufnahme der Umgebungsgeräusche aufgezeichnet. Die Beschleunigungswerte werden fast in allen Arbeiten zur Aktivitätserkennung verwendet. Durch die Verwendung des Luftdrucks soll die Erkennung von Stockwerkswechseln in den U-Bahn-Stationen ermöglicht werden. Die Audiodaten wiederum sollen primär die Unterscheidung verschiedener Bewegungsphasen der U-Bahn selbst unterstützen.
Der Ablauf des Verfahrens besteht aus den folgenden Schritten:

- Vorverarbeitung / Aufbereitung der Daten
- Bildung von Fenstern
- Berechnung der Features
- Klassifikation der Daten mit Hilfe eines Algorithmus des maschinellen Lernens

Diese werden im Folgenden einzeln erläutert.

Vorverarbeitung / Aufbereitung der Daten Die Hardware-Sensoren aktueller Smartphones weisen in der Regel Abtastraten zwischen 20 und 80 Hz auf. Allerdings unterliegt die Abtastung der einzelnen Sensoren gewissen Schwankungen, sodass in einem Vorverarbeitungsschritt die Zeitserien der Messwerte der Sensoren künstlich in eine äquidistante Form gebracht werden müssen. Diese Angleichung erleichtert die spätere Berechnung der Features.

Fensterbildung Die Aktivitätserkennung wird in diesem Fall nicht auf den rohen Sensordaten durchgeführt, sondern wie in der Literatur üblich, auf über bestimmte Zeitintervalle aggregierte Daten. Die Zeitreihe wird also in Fenster einer bestimmten Länge unterteilt, die dann jeweils durch Berechnung bestimmter Features auf einzelne Werte aggregiert werden.
In den verwandten Arbeiten findet sich eine große Bandbreite der verwendeten Fensterlängen. Ansätze, die sich mit der Erkennung des verwendeten Verkehrsmittels beschäftigen, verwenden meist längere Fenster von acht oder mehr Sekunden [185]. Rein auf körperliche Aktivitäten bezogene Ansätze nutzen eher kürzere Fenster im Bereich von ein oder zwei Sekunden [201]. Aufgrund des Ziels der feingranularen Aktivitätserkennung, die zudem mehr der Erkennung körperlicher Aktivitäten als der groben Erkennung des Transportmittels ähnelt, wurden für das vorgestellte Verfahren Fenstergrößen von $1024ms$, $2048ms$ und $4096ms$ gewählt. Die Wahl von Zweierpotenzen ist ein übliches Vorgehen [93] und erleichtert die spätere Anwendung einer Fast-Fourier-Transformation. Die Fenster werden so gebildet, dass sie sich um 50% überlappen.

Berechnung der Features Auf den so gebildeten Fenstern werden im Anschluss Features berechnet, welche die im Fenster auftretenden Werte auf unterschiedliche Arten aggregieren. Im Rahmen dieses Verfahrens werden sowohl Features im Ortsraum als auch – nach Anwendung einer Fourier-Transformation – im Frequenzraum berechnet.
Für den dreiachsigen Beschleunigungssensor werden die Werte vorab durch Berechnung der Norm anhand der Formel $a = \sqrt{a_x^2 + a_y^2 + a_z^2}$ auf einen lageunabhängigen Wert abgebildet, da wie in den Anforderungen definiert, die Trageposition des Smartphones frei wählbar ist. Für die so vorberechneten Werte sowie für die Werte des Barometers werden dann für jedes Fenster statistische Maße wie das Maximum, das Minimum, das arithmetische Mittel, das quadratische Mittel, die Standardabweichung, das 75-Prozent-Quantil und die Anzahl der Nulldurchgänge berechnet. Für jede dieser Kombinationen aus Sensor und Maß wird zudem die Differenz zum entsprechenden Feature jedes der vorherigen zehn Fenster sowie des nachfolgenden Fensters berechnet. Die Auswertung eines Zeitfensters verläuft also um eine Fensterlänge verzögert, um auch das nachfolgende Fenster berücksichtigen zu können.
Die Zeitreihe jeder dieser beiden Komponenten sowie die bisher nicht behandelten Audiodaten werden dann mit Hilfe einer Fast-Fourier-Transformation (FFT) in den Frequenzraum überführt. In diesem werden die bereits erwähnten

statistischen Maße sowie die Energie, definiert als

$$\frac{1}{n}\sum_{i=1}^{n} {x_i}^2 \tag{6.1}$$

und Entropie, definiert als

$$-\sum_{i=1}^{n} P(x_i) log_b P(x_i) \tag{6.2}$$

berechnet.
Die Berechnungen im Frequenzraum werden dabei nicht nur auf dem gesamten Frequenzspektrum, sondern auch nochmals unterteilt in einzelne Frequenzbänder durchgeführt. Für die Daten des Beschleunigungssensors und des Barometers werden dabei die Frequenzbänder

1-3, 3-5, 5-8, 8-11, 11-16, 16-22, 22-29, 29-37 und 37-50 Hz

verwendet und für die Audiodaten die Bänder

1-20, 20-50, 50-100, 100-200, 200-500, 500-900, 900-1400, 1400-2000, 2000-2700, 2700-4000, 1-500, 500-1500 und 1500-4000 Hz

Für diese einzelnen Frequenzbänder wird abschließend noch das jeweilige Verhältnis von Durchschnitt, Energie und Entropie des einzelnen Frequenzbandes zum selben Maß des gesamten Spektrums berechnet.

Die umfangreiche Verwendung von audio-basierten Features wurde aufgrund von Beobachtungen, die in vorhergehenden Testaufnahmen gemacht wurden, gewählt. In Abbildung 6.2 ist ein Spektrogramm abgebildet, das die Frequenzverteilung in der Audioaufnahme während einer U-Bahn-Fahrt darstellt. Auf der Ordinate sind die Frequenzen aufgetragen, während auf der Abszisse der zeitliche Verlauf zu sehen ist. Die farbliche Kodierung entspricht der Energie der entsprechenden Frequenz zum jeweiligen Zeitpunkt. Je röter, umso präsenter ist die entsprechende Frequenz im Spektrum.
Beim Vergleich mit der originalen Audioaufnahmen konnten einige interessante Aspekte identifiziert werden. Zwischen Zeitpunkt (1) und (2) kann man im Bereich von ca. 2,5 kHz mehrere Signalpeaks erkennen. Diese wurden durch den zu diesem Zeitpunkt abgespielten Warnton der U-Bahn, der das Schließen der Türen begleitet, ausgelöst. Nach diesen Signaltönen schließen die Türen, was durch eine große Energie über den gesamten Frequenzbereich zu erkennen ist. Dieses „Rauschen“ entsteht durch die pneumatische Schließvorrichtung der Wagon-Türen [1].
Die Zeit danach, während die U-Bahn noch steht, die Türen aber geschlossen sind, zeichnet sich durch generell weniger Energie im gesamten Spektrum, also eine größere Stille aus. Bei Zeitpunkt (3) fährt die U-Bahn los und beschleunigt. Sehr auffällig ist hier der annähernd halbkreis-förmige Verlauf der

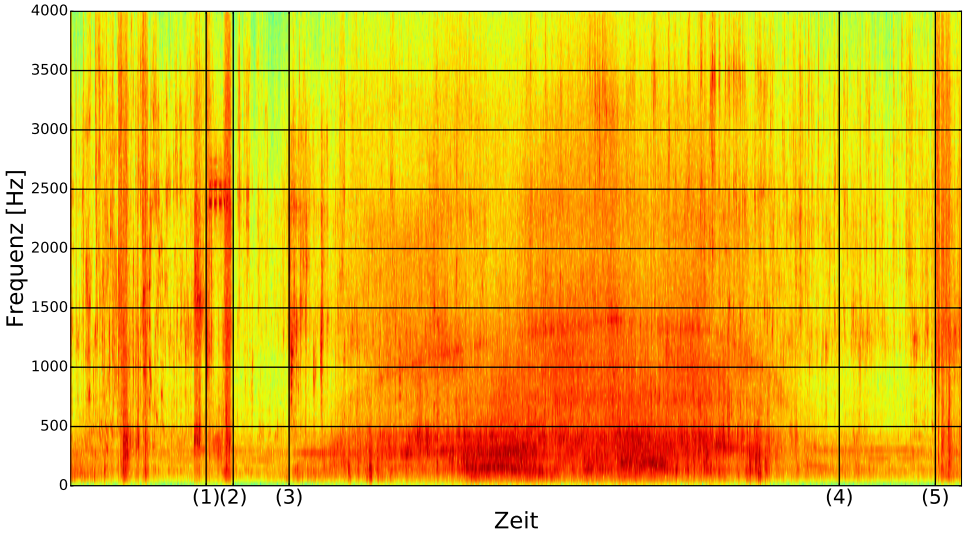


Abbildung 6.2: Spektrogramm, das die Frequenzverteilung der Audio-Daten während einer U-Bahn-Fahrt darstellt.

energiereichen Frequenzen, speziell zwischen 1,0 und 1,5 Hz, der exakt zu den Zeiträumen von Beschleunigung, Fahrt und Bremsen passt.

Ab ungefähr Zeitpunkt (4) folgt wieder eine Phase der Stille, in der die U-Bahn bereits zum Stehen gekommen ist, die Türen aber noch geschlossen sind. Bei (5) öffnen sich schließlich die Türen, was sich wieder durch eine Signalspitze aufgrund der Tür-Pneumatik sowie die generell höhere Energie bemerkbar macht.

Um den sehr charakteristischen Signalton noch explizit in den berechneten Features abbilden zu können, wird für das Frequenzband von 2,4 bis 2,6 kHz zusätzlich zu den bereits vorhandenen Audio-Features noch das Maximum, das arithmetische Mittel, das quadratische Mittel sowie die Energie berechnet, und zwar ausgehend vom aktuellen Fenster für die vorherigen n Fenster ($n \in 5, 10, 20, 30$), und als Features des aktuellen Fensters hinzugefügt. Es soll damit also festgehalten werden, ob in der näheren Vergangenheit des aktuellen Fensters der Signalton zu hören war.

Insgesamt ergeben sich daraus 632 verschiedene Features die zur Erkennung verwendet werden können. Da diese hohe Anzahl sowohl im Hinblick auf die Performanz als auch bzgl. der Klassifikationsgenauigkeit kontraproduktiv ist [201], wurde im Rahmen der Evaluation mit Hilfe der *Correlation-Based Feature Subset Selection* [84] eine Untermenge von nur noch 80 relevanten Features extrahiert. Es müssen daher nicht alle beschrieben Features tatsächlich berechnet werden, jedoch sind in der finalen Menge Features aus allen beschriebenen Klassen enthalten.

Klassifikation durch maschinelles Lernen Die berechneten Features werden schlussendlich als Eingabedaten zur Klassifikation mit Hilfe eines Algorithmus des maschinellen Lernens verwendet. Im Rahmen der Evaluation wurden verschiedene Algorithmen getestet.

6.4 Evaluation der feingranularen Aktivitätserkennung

Das vorgeschlagene Verfahren wurde mit Hilfe eines realen Datensatzes evaluiert. Im Folgenden werden die Erkenntnisse bzgl. Klassifikationsgüte sowie des Einflusses unterschiedlicher Parameter wie z.B. der Trageposition des Endgerätes erläutert.

6.4.1 Versuchsaufbau und Datensatz der Evaluation

Der zur Evaluation verwendete Datensatz wurde von Uwe Müller in seiner, vom Autor dieser Arbeit betreuten, Abschlussarbeit [141] gesammelt und vom Autor dieser Arbeit zur Evaluation des im vorherigen Abschnitts vorgestellten Verfahrens verwendet.
Der Datensatz besteht aus insgesamt 279 Minuten aufgezeichneter Aktivitäten im U-Bahn-Netz der Stadt München. Eine Testperson wurde dafür mit insgesamt vier Android-Smartphones [70] vom Typ Nexus 4 [73] des Herstellers LG [116] ausgestattet. Die Smartphones wurden an vier unterschiedlichen Positionen getragen:

- In der linken vorderen Hemdtasche
- In der rechten vorderen Hosentasche
- Frei liegend im Rucksack
- In der rechten Hand gehalten

Auf den Smartphones wurde eine Software zur Aufzeichnung der benötigten Sensordaten installiert. Zusätzlich trug die Testperson ein weiteres, unabhängiges Smartphone bei sich, mit dem sie die durchgeführten Aktivitäten vermerken („labeln“) konnte. Die Datenaufzeichnungs-Smartphones waren über ein WLAN-Ad-Hoc-Netz mit dem Label-Smartphone verbunden, welches entsprechende Labels direkt beim Erstellen per Broadcast an die anderen Smartphones verteilt hat. Die Aufzeichnung bzw. das Labeln war dadurch nahezu perfekt synchronisiert ohne eine synchrone Uhr auf den Smartphones zu erfordern.
Die Testperson bewegte sich zur Datenaufzeichnung im U-Bahn-Netz der Stadt München und führte alle in Tabelle 6.1 aufgeführten Aktivitäten mehrfach durch. Bedingt durch das realistische Testszenario sind die Häufigkeiten der

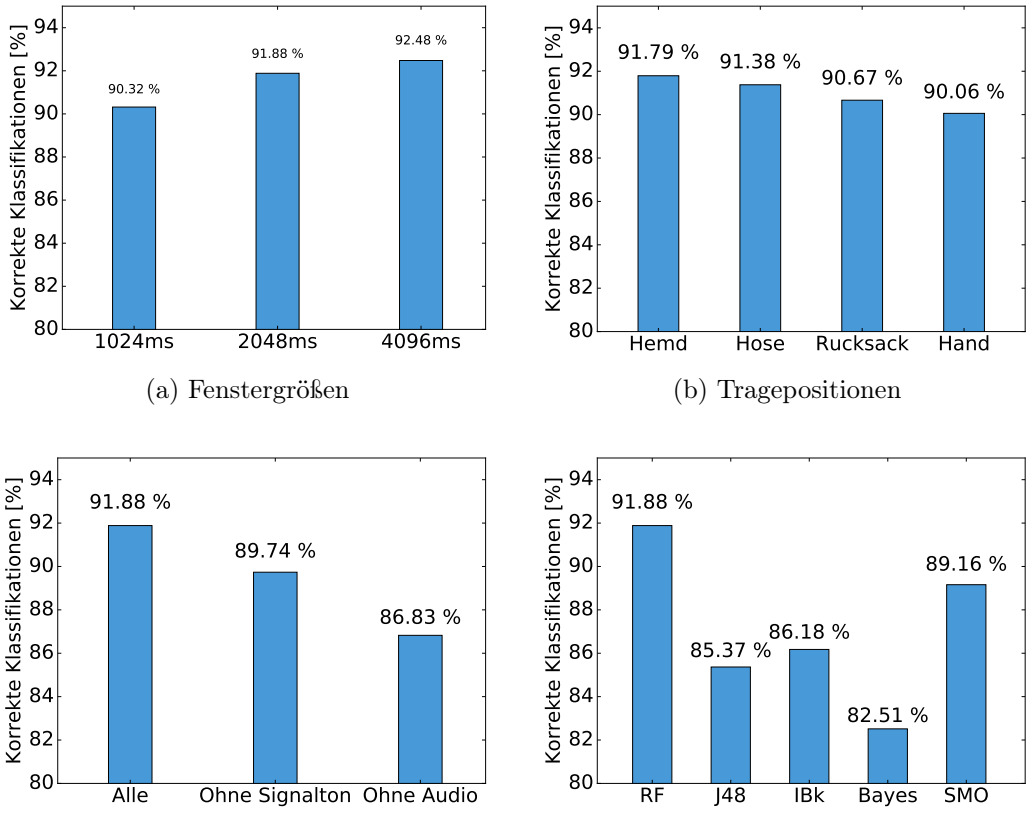
Aktivitäten nicht gleichmäßig verteilt. Das häufigste Aktivität („gehen“) umfasst 32,9% der Aktivitäten. Für die spätere Evaluation ist dies wichtig, da dadurch eine untere Grenze der Erkennungsgenauigkeit festgelegt ist, die allein durch „Raten“ des häufigsten Labels erreicht werden konnte.
Bei der im Verfahren vorgesehenen Fensterbildung können durch die Festlegung der Intervalle Fenster mit mehr als einem Label entstehen. In diesen Fällen wurde dem Fenster das Label zugeordnet, das die längste Zeitspanne des Fensters gültig war.

6.4.2 Ergebnisse der Evaluation auf dem erstellten Datensatz

Zur Evaluation der Erkennungsrate des vorgeschlagenen Verfahrens wurde die WEKA data mining software [83] verwendet. Sofern nicht anders angegeben wurde zur Klassifizierung der Aktivitäten der Random-Forest-Algorithmus [30] eingesetzt. Die Ergebnisse wurden erhalten, indem eine 10-fache Kreuzvalidierung auf dem Datensatz durchgeführt wurde. In Abbildung 6.3 sind die Resultate bzgl. verschiedener Gesichtspunkte dargestellt.

Fenstergrößen Zunächst wurde der Einfluss der Fenstergrößen auf die Erkennungsgenauigkeit ausgewertet. Wie in Abbildung 6.3a zu sehen, unterscheiden sich die Erkennungsraten der drei getesteten Größen nur minimal, wobei eine Verbesserung der Genauigkeit mit zunehmender Fenstergröße von 90,32% bei 1024ms hin zu 92,48% bei 4096 ms beobachtet werden kann. Da, wie im vorherigen Abschnitt zur Proximitätserkennung durch Aktivitätssequenzen beschrieben, eine größere Anzahl an Aktivitätswechseln vorteilhaft für die Proximitätserkennung ist, sind zu lange Fenster zu vermeiden, da darin einzelne Aktivitätswechsel „verschluckt“ werden könnten. Aus diesem Grund wurde für die weiteren Auswertungen als Kompromiss aus Fensterlänge und Klassifikationsgenauigkeiten die Fenstergröße 2048ms mit einer Erkennungsgenauigkeit von 91,88% verwendet.
Allgemein kann an dieser Stelle schon festgehalten werden, dass mit einer Klassifikationsgenauigkeit von über 90% ein sehr gutes Ergebnis erzielt werden konnte.

Tragepositionen Die Auswertung zu den Fenstergrößen wurde auf dem gesamten Datensatz, d.h. kombiniert aus den Daten aller vier Test-Smartphones durchgeführt. Um den Einfluss unterschiedlicher Tragepositionen zu beurteilen, wurde für eine weitere Auswertung jeder Datensatz einzeln betrachtet. Die Ergebnisse sind in Abbildung 6.3b dargestellt.
Auch in diesem Fall sind die Unterschiede nur marginal, mit mindestens 90,06% und maximal 91,79% richtig erkannter Aktivitäten. Es ist jedoch trotzdem ein leicher Abwärtstrend festzustellen von Tragepositionen, in denen das Smartphone stärker fixiert ist (Hemdtasche und Hosentasche) hin zu denen, in de-



(a) Fenstergrößen

(b) Tragepositionen

(c) Berücksichtigte Features

(d) Klassifkationsalgorithmen

Abbildung 6.3: Anteil richtig klassifizierter Instanzen, abhängig von (a) der Fenstergröße, (b) der Trageposition des Endgerätes, (c) der verwendeten (Unter-)Menge an Features und (d) dem verwendeten Klassifizierungsalgorithmus.

nen es stärkeren Schwankungen ausgesetzt ist (Rucksack und Hand). Dieser Trend erscheint durch das zunehmende „Rauschen“ bei weniger stark fixierter Trageposition logisch. Erfreulich ist trotzdem die generelle hohe Klassifikationsgenauigkeit über alle Tragepositionen hinweg.

Einfluss der Audio-Features Da insbesondere die Audio-Features gezielt ausgewählt wurden, um dem U-Bahn-Szenario Rechnung zu tragen, wurde deren Einfluss ebenfalls gesondert evaluiert. Hierzu wurden drei verschiedene Feature-Mengen festgelegt: Alle wie bisher, alle ohne die dediziert dem Signalton gewidmeten Audio-Features und alle ohne sämtliche Audio-Features. Die Resultate sind in Abbildung 6.3c aufgeführt.
Lässt man nur die signaltonbezogenen Audio-Features weg, so verschlechtert sich das Ergebnis von 91,88% auf 89,74%. Ein Unterschied, der den Aufwand für die Berechnung der Audio-Features evtl. noch nicht rechtfertigen würde. Verzichtet man jedoch komplett auf die Audio-Features, so gibt es einen deut-

licheren Einbruch auf 86,83%. Da davon weniger die generischen Aktivitäten wie „gehen“ sondern insbesondere die wichtigen U-Bahn-bezogenen Aktivitäten wie „in die U-Bahn einsteigen“ betroffen sind, ist dieser Abfall der Erkennungsgenauigkeit umso drastischer.

Klassifikationsalgorithmen Wie bereits erwähnt, wurde für die Auswertungen der Random-Forest-Klassifizierungsalgorithmus verwendet. Es wurden alternativ noch weitere Algorithmen evaluiert. Dabei handelt es sich um einfache Entscheidungsbäume, k-Nächste-Nachbarn, bayesische Netze und Support-Vector-Machines. Die Ergebnisse sind in Abbildung 6.3d zu sehen.
Der Random-Forest-Algorithmus schneidet dabei mit teilweise großem Abstand am besten ab. Sein Vorteil liegt hauptsächlich jedoch darin, dass er nicht so stark von Parametereinstellungen abhängig ist wie andere Algorithmen. Er ist zudem weniger anfällig für Overfitting. Es ist sehr wahrscheinlich, dass bei geeigneterer Parameterwahl auch die anderen Algorithmen bessere Ergebnisse, auch besser als der Random-Forest-Algorithmus, aufweisen können.
Ziel der Evaluation war es jedoch, die grundsätzliche Machbarkeit einer so feingranularen Aktivitätserkennung im U-Bahn-Szenario zu zeigen. Dafür werden die sehr guten Werte des Random-Forest-Algorithmus als ausreichend erachtet.

Zusammenfassend lässt sich also festhalten, dass durch geeignete Wahl der Features, ein sehr gutes Klassifikationsergebnis erreicht werden kann. Damit erscheint eine Proximitätserkennung auf Basis feingranularer Aktivitätssequenzen nicht nur theoretisch, sondern auch praktisch machbar.

6.5 Diskussion und Zusammenfassung

In diesem Kapitel wurde zunächst gezeigt, dass eine Proximitätserkennung rein auf Basis von Aktivitätssequenzen grundsätzlich möglich ist. Mit zunehmender Länge werden solche Sequenzen auch in großen Gesamtpopulationen von mehreren zehntausend Menschen immer eindeutiger. Gleichzeitig wurde jedoch offensichtlich, dass mit nur einer einfachen Aktivitätserkennung ein sehr langer Zeitraum notwendig ist, um eine (nahezu) Eindeutigkeit und damit sinnvolle Proximitätserkennung zu erreichen. Mit einer deutlich feigranulareren Erkennung ließen sich diese Zeiträume deutlich verkürzen.
Deswegen wurde im weiteren Verlauf anhand eines U-Bahn-Szenarios gezeigt, dass eine derart feingranulare Aktivitätserkennung tatsächlich mit standardmäßigen mobilen Endgeräten möglich ist. Hierzu wurden geeignete Features auf den Sensor- und Audiodaten eines Smartphones berechnet und diese dann mit Algorithmen des maschinellen Lernens zur Klassifikation der Aktivitäten verwendet. Es war damit möglich, 17 verschiedene Aktivitäten zuverlässig zu unterscheiden.
Ein Proximitätserkennungsverfahren auf Basis solcher Aktivitätssequenzen erfüllt grundsätzlich die primären Anforderungen bzgl. Fälschungssicherheit und

Schutz der Privatsphäre. Die auftretenden Aktivitätssequenzen sind ausreichend zufällig um nicht mit vertretbarem Aufwand vorhersagbar zu sein. Die Sequenzen an sich enthalten zudem keinerlei Ortsinformation und sind für sich selbst aus Privatsphäre-Sicht relativ unkritisch. In zukünftigen Arbeiten sollte jedoch untersucht werden, ob sich die Aktivitätssequenzen auf bestimmten Streckenabschnitten soweit ähneln, dass eine Berechnung des entsprechenden Streckenabschnitts möglich ist. Dies wäre ein Angriffspunkt, um die Privatsphäre zu beeinträchtigen. Analog zum ProbeTag- und zum SURFtogether-Verfahren ließe sich dieses Problem jedoch dann mit Hilfe eines PPT-Verfahrens beheben.
Die Berechnung der Aktivitäten selbst ist trotz der Verwendung einer Vielzahl von Sensoren und Daten aus Privatsphäresicht – bezogen auf das System- und Angreifermodell aus Abschnitt 2.6 – unkritisch, da sämtliche Daten lokal erhoben und verarbeitet werden und nur die aggregierten Aktivitäten das Endgerät verlassen. Das Verfahren ist zudem insgesamt wie gefordert auf standardmäßigen Endgeräten umsetzbar, es benötigt keine dedizierte Infrastruktur und es bezieht sich per Definition nicht rein auf eine örtliche, sondern durch die Verwendung der Aktivitäten auf eine kontextuelle Proximität.

7 Zusammenfassung und Ausblick

In der vorliegenden Arbeit standen die kontextbezogenen Dienste und insbesondere solche, die sich die Proximität eines Benutzers zu anderen Benutzern oder Objekten zu Nutze machen wollen, als primäre Motivation zur Entwicklung verschiedener Verfahren im Mittelpunkt. Während eine grundsätzliche Proximitätserkennung mit Hilfe aktueller Technologien wie GPS-Ortung problemlos möglich ist, leiden alle diese einfachen Verfahren unter erheblichen Schwächen bzgl. Sicherheit und Privatsphäre der Nutzer. Zudem beschränken sie sich rein auf räumliche Proximität. Eine darüberhinausgehende kontextuelle Proximität, die z.B. noch die Aktivitäten der Benutzer berücksichtigt, wird nicht erkannt.

Um diese Herausforderungen zu lösen, wurden in der vorliegenden Arbeit drei sichere und privatsphäre-schonende Verfahren vorgestellt, die sich in unterschiedlichem Maße auf die räumliche bzw. kontextuelle Proximität zwischen Entitäten beziehen und diese erkennen.

Als erstes wurde das ProbeTag-Verfahren vorgestellt. Dieser Ansatz basiert auf der Aufzeichnung von WLAN-Management-Frames, die von mobilen Endgeräten in der Umgebung ausgesandt werden, und dem Vergleich dieser zur Bestimmung einer Ähnlichkeit. Dadurch, dass sich die Zusammensetzung der Personen und damit der mobilen Endgeräte an einem Ort kontinuierlich ändert und es extrem unwahrscheinlich ist, dass sich eine solche Zusammensetzung in der gleichen Form am gleichen Ort wiederholt, sind die so erstellten Location Tags orts- und zeitspezifisch und können damit weder vorhergesagt noch aus historischen Daten wiedereingespielt werden. In einer umfangreichen Evaluation wurde zunächst gezeigt, dass das vorgeschlagene Verfahren eine grundsätzliche Proximität zwischen zwei Endgeräten erkennen kann, und dies sowohl in statischen Szenarien als auch in sehr realistischen, in denen sich die beteiligten Benutzer in einer realen Umgebung zusammen fortbewegen. Darüberhinaus besitzt das Verfahren auch grundsätzliche Eigenschaften zur Abschätzung einer relativen Position. Durch einen Langzeittest wurde zudem auch experimentell gezeigt, dass die konstruierten ProbeTags nicht nur orts- sondern auch zeitspezifisch sind und damit sehr gute Voraussetzungen als Kernelement eines Proximitätserkennungsverfahrens besitzen.

Während das ProbeTag-Verfahren rein auf räumliche Nähe abzielt, berücksichtigt das zweite vorgestelle Verfahren implizit auch die Aktivitäten der Nutzer. Das sogenannte SURFtogether-Verfahren basiert auf der Grundidee, dass Per-

sonen, die sich in einer Gruppe, d.h. hoher Proximität, befinden, die meiste Zeit die selben Personen und Objekte sehen. Mit Hilfe von am Körper getragenen Kameras wie Smart Glasses oder Action Kameras kann das Blickfeld der Benutzer kontinuierlich ausgewertet werden. Das SURFtogether-Verfahren abstrahiert die gesehenen Bildauschnitte mit Hilfe des SURF-Verfahrens und benutzt die so gewonnen Merkmalsvektoren zur Proximitätsschätzung. Diese visuellen Location Tags sind stark abhängig von sich bewegenden Objekten und damit analog zu den ProbeTags orts- und zeitspezifisch. Darüberhinaus spielt die Aktivität der Benutzer eine Rolle, da Benutzer, die sich zwar am gleichen Ort befinden, jedoch unabhängig voneinander agieren, seltener ähnliche Bildausschnitte aufzeichnen werden. Das SURFtogether-Verfahren zielt also auf die Erkennung von Gruppensituationen ab. In einer umfangreichen Evaluation wurden diverse Szenarien sowie Erweiterungen des Verfahrens untersucht, sowohl in einer kontrollierten Umgebung in Form einer 3D-Simulation als auch in einem realen Testlauf. Unter Einbeziehung einer Logik-Ebene, welche die rohen Proximitätsschätzungen auswertet und ein geglättetes Ergebnis produziert, konnten sehr gute Klassifikationsergebnisse erzielt werden.

Das dritte vorgestellte Verfahren geht noch einen Schritt weiter und basiert auf der Idee, eine Proximität rein anhand von Aktivitäten bzw. Aktivitätssequenzen zu schätzen. Hierzu wurde zunächst in einer Simulation gezeigt, dass auch in einer großen Gesamtpopulation von mehreren zehntausend Entitäten mit ausreichend großen Zeitfenstern, d.h. Sequenzlängen, einzelne Individuen sehr gut anhand der Aktivitätssequenzen identifiziert werden können. Die benötigten Zeitfenster sind jedoch im Falle aktueller Standard-Aktivitätserkennung mit nur wenigen Aktivitäten zu lang für einen sinnvollen Einsatz. Demgegenüber kann mit einer sehr feingranularen Aktivitätserkennung mit relativ kurzen Zeitfenstern ein gutes Ergebnis erzielt werden. Dass eine solch feingranulare Aktivitätserkennung mit aktuellen mobilen Endgeräten umgesetzt werden kann, wurde in einem zweiten Schritt gezeigt. Mit Hilfe einer geeigneten Feature-Auswahl und Algorithmen des maschinellen Lernens konnten relativ zuverlässig 17 verschiedene Aktivitäten unterschieden werden, die bei der Fortbewegung im öffentlichen Personennahverkehr, im Speziellen mit der U-Bahn, auftreten können.

Die vorgestellten Verfahren erfüllen die grundsätzlich gestellten Anforderungen: Sie sind auf Standard-Hardware umsetzbar, sie benötigen keine dedizierte Infrastruktur, sie haben ihren Fokus auf Proximitäten im Nahbereich von bis zu 10 Metern, d.h. auf Gruppensituationen, und sie berechnen zum Teil nicht nur die räumliche Proximität sondern auch eine kontextuelle, d.h. insbesondere eine die Aktivitäten berücksichtigende Proximität.

Die zur Proximitätsbestimmung verwendeten Informationseinheiten, d.h. die ProbeTags, SURFtogether-Tags oder die Aktivitätssequenzen, weisen alle keine dedizierte Ortssemantik auf und sind durch das verwendete Datenmaterial weitestgehend unfälschbar. Durch diese Eigenschaften erfüllen die drei Ver-

fahren auch die gestellten Anforderungen bzgl. Sicherheit und Privatsphäre. Letztere könnte jedoch theoretisch durch die Verwendung einer vorher aufgezeichneten Datenbank beeinträchtigt werden. Zwar sind die Informationseinheiten wie beschrieben nicht fälschbar, jedoch reicht es in manchen Fällen aus, wenn auch nur eine geringe Ähnlichkeit mit einem vorhandenen (älteren) Datenbankeintrag besteht, um den absoluten Aufenthaltsort (grob) zu schätzen. Dieses Problem könnte in zukünftigen Arbeiten auf verschiedenen Wegen gelöst werden. Eine Möglichkeit besteht darin, insbesondere beim ProbeTag und beim SURFtogether-Verfahren lokal eine Bewertung der betrachteten Daten (WLAN-Signale bzw. Bilder) vorzunehmen, und solche Elemente, die als sehr statisch erkannt werden, herauszufiltern. Damit bleiben nur die komplett dynamischen Elemente übrig, was ein Matching mit einer vorhandenen Datenbank unmöglich machen würde.
Alternativ dazu bietet sich die Kombination mit einem Verfahren aus dem Bereich des PPT an. Letztere können sehr gut sicherstellen, dass die Eingabedaten der einzelnen Teilnehmer des Proximitätstests nicht offen gelegt und mit einer Datenbank verglichen werden, sie sind jedoch im Gegenzug anfällig für gefälschte Eingabedaten. Dieses Problem wiederum wird durch die vorgestellten Verfahren hevorragend gelöst. Damit erscheint eine Kombination dieser beiden Bereiche als eine der vielversprechendsten Möglichkeiten zur sicheren und privatsphäreschonenden Erkennung von kontextueller Proximität.

Abkürzungsverzeichnis

ADL Activities of Daily Living

AP Access-Point

BeiDou-2 BeiDou Navigation Satellite System

FFT Fast-Fourier-Transformation

FIFO First-in-First-out

GLONASS Globalnaya navigatsionnaya sputnikovaya sistema

GNSS Global Navigation Satellite System

GPS Global Positioning System

IPt Interest Point

LAC Location Area

LBS Location Based Service

LTE Long-Term Evolution

MCC Matthews-Korrelations-Koeffizient

OSN Online Social Network

P2P Peer-to-Peer

P3P Platform for Privacy Preferences Project

PET Private Equality Testing

PIR Private Information Retrieval

PPT Private Proximity Testing

PEC Proximitäts-Erkennungs-Client

POI Point-of-Interest

PSI Private Set Intersection

PTS Proximity-Test-Dienst

SSID Service Set Identification

ST-Tag SURFtogether-Tag

SWE Sektorbezogene Warteschlangen-Erweiterung

TTP Trusted Third Party

W3C World Wide Web Consortium

WLAN Wireless Local Area Network

WWW World Wide Web

Literaturverzeichnis

[1] Abendzeitung München: *U-Bahn-Türen - Nur mit grober Gewalt*. http://www.abendzeitung-muenchen.de/inhalt.so-funktionieren-sie-u-bahn-tueren-nur-mit-grober-gewalt.2ff22c48-7a4f-449b-b03a-f851bd7d5a69.html, 2016. letzer Abruf: 10.01.2016.

[2] Abowd, G. D., A. K. Dey, P. J. Brown, N. Davies, M. Smith und P. Steggles: *Towards a better understanding of context and context-awareness*. In: *Handheld and ubiquitous computing*, Bd. 1707 d. Reihe *Lecture Notes in Computer Science*, S. 304–307. Springer, 1999.

[3] Abul, O., F. Bonchi und M. Nanni: *Never walk alone: Uncertainty for anonymity in moving objects databases*. In: *Proceeedings of the 24th IEEE International Conference on Data Engineering (ICDE 2008)*, S. 376–385. IEEE, 2008.

[4] Addlesee, M., R. Curwen, S. Hodges, J. Newman, P. Steggles, A. Ward und A. Hopper: *Implementing a sentient computing system*. IEEE Computer, 34(8):50–56, 2001.

[5] AeroScout Industrial: *Aeroscout*. http://www.aeroscout.com/, 2016. letzer Abruf: 07.01.2016.

[6] Amir, A., A. Efrat, J. Myllymaki, L. Palaniappan und K. Wampler: *Buddy tracking – efficient proximity detection among mobile friends*. Pervasive and Mobile Computing, 3(5):489–511, 2007.

[7] Apple: *CMMotionActivity*. https://developer.apple.com/library/ios/documentation/CoreMotion/Reference/CMMotionActivity_class/index.html, 2016. letzer Abruf: 10.01.2016.

[8] Apple: *iBeacon*. https://support.apple.com/en-gb/HT202880, 2016. letzer Abruf: 07.01.2016.

[9] Apple: *Watch*. http://www.apple.com/watch/, 2016. letzer Abruf: 17.01.2016.

[10] Apple App Store: *Anomo - Meet New People*. https://itunes.apple.com/us/app/anomo-meet-new-people/id529027583, 2016. letzer Abruf: 17.01.2016.

[11] Apple App Store: *Highlight - Meet New People, Find and Connect with Friends Nearby*. https://itunes.apple.com/us/app/highlight-meet-new-people/id441534409, 2016. letzer Abruf: 17.01.2016.

[12] Apple App Store: *SocialRadar - Find your friends on a map*. https://itunes.apple.com/us/app/socialradar-find-your-friends/id720159037, 2016. letzer Abruf: 17.01.2016.

[13] Aviv, A. J., B. Sapp, M. Blaze und J. M. Smith: *Practicality of accelerometer side channels on smartphones*. In: *Proceedings of the 28th Annual Computer Security Applications Conference (ACSAC 2012)*, S. 41–50. ACM, 2012.

[14] Bahl, P. und V. Padmanabhan: *RADAR: An In-Building RF-based User Location and Tracking System*. In: *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000)*, Bd. 2, S. 775–784. IEEE, 2000.

[15] Bamba, B., L. Liu, P. Pesti und T. Wang: *Supporting anonymous location queries in mobile environments with privacygrid*. In: *Proceedings of the 17th international conference on World Wide Web*, S. 237–246. ACM, 2008.

[16] Bao, L. und S. S. Intille: *Activity recognition from user-annotated acceleration data*. In: *Proceedings of the 2nd International Conference on Pervasive Computing (PERVASIVE 2004)*, Bd. 3001 d. Reihe *Lecture Notes in Computer Science*, S. 1–17. Springer, 2004.

[17] Barbera, M. V., A. Epasto, A. Mei, V. C. Perta und J. Stefa: *Signals from the crowd: uncovering social relationships through smartphone probes*. In: *Proceedings of the 2013 Conference on Internet Measurement (IMC 2013)*, S. 265–276. ACM, 2013.

[18] Bay, H., A. Ess, T. Tuytelaars und L. Van Gool: *Speeded-Up Robust Features (SURF)*. Computer Vision and Image Understanding, 110(3):346–359, 2008.

[19] Bay, H., T. Tuytelaars und L. Van Gool: *SURF: Speeded Up Robust Features*. In: *Proceedings of the 9th European Conference on Computer Vision (ECCV 2006)*, Bd. 3951 d. Reihe *Lecture Notes in Computer Science*, S. 404–417. Springer, 2006.

[20] Bell, D. E. und L. J. La Padula: *Secure computer system: Unified exposition and multics interpretation*. Techn. Ber., DTIC Document, 1976.

[21] Beresford, A. R. und F. Stajano: *Location privacy in pervasive computing*. IEEE Pervasive computing, 2(1):46–55, 2003.

[22] Berners-Lee, T. und R. Cailliau: *WorldWideWeb: Proposal for a HyperText Project*. `https://www.w3.org/Proposal.html`, 1990. letzer Abruf: 17.01.2016.

[23] Bird, C., A. Gourley, P. Devanbu, M. Gertz und A. Swaminathan: *Mining email social networks*. In: *Proceedings of the 2006 international workshop on Mining software repositories (MSR 2006)*, S. 137–143. ACM, 2006.

[24] Bonné, B., A. Barzan, P. Quax und W. Lamotte: *WiFiPi: Involuntary tracking of visitors at mass events*. In: *Proceedings of the 14th IEEE International Symposium and Workshops on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2013)*, S. 1–6. IEEE, 2013.

[25] Bottazzi, D., R. Montanari und A. Toninelli: *Context-Aware Middleware for Anytime, Anywhere Social Networks*. IEEE Intelligent Systems, 22(5):23–32, 2007.

[26] Boudot, F., B. Schoenmakers und J. Traore: *A fair and efficient solution to the socialist millionaires' problem*. Discrete Applied Mathematics, 111(1–2):23–36, 2001.

[27] Brands, S. und D. Chaum: *Distance-Bounding Protocols*. In: *Proceedings of the 1993 Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT 1993)*, Bd. 765 d. Reihe *Lecture Notes in Computer Science*, S. 344–359. Springer, 1994.

[28] Brdiczka, O., J. Maisonnasse und P. Reignier: *Automatic detection of interaction groups*. In: *Proceedings of the 7th international conference on Multimodal interfaces (ICMI 2005)*, S. 32–36. ACM, 2005.

[29] Breiman, L.: *Bagging predictors*. Machine learning, 24(2):123–140, 1996.

[30] Breiman, L.: *Random forests*. Machine learning, 45(1):5–32, 2001.

[31] Buttyán, L., T. Holczer und I. Vajda: *On the Effectiveness of Changing Pseudonyms to Provide Location Privacy in VANETs*. In: *Proceedings of the 4th European Workshop on Security and Privacy in Ad-hoc and Sensor Networks (ESAS 2007)*, Bd. 4572 d. Reihe *Lecture Notes in Computer Science*, S. 129–141. Springer, 2007.

[32] Calonder, M., V. Lepetit, C. Strecha und P. Fua: *BRIEF: Binary Robust Independent Elementary Features*. In: *Proceedings of the 11th European Conference on Computer Vision (ECCV 2010)*, Bd. 6314 d. Reihe *Lecture Notes in Computer Science*, S. 778–792. Springer, 2010.

[33] Chartrand, T. L. und J. A. Bargh: *The chameleon effect: the perception – behavior link and social interaction*. Journal of personality and social psychology, 76(6):893–910, 1999.

[34] Chon, Y., S. Kim, S. Lee, D. Kim, Y. Kim und H. Cha: *Sensing WiFi Packets in the Air: Practicality and Implications in Urban Mobility Monitoring*. In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2014)*, S. 189–200. ACM, 2014.

[35] Chor, B. und N. Gilboa: *Computationally private information retrieval*. In: *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, S. 304–313. ACM, 1997.

[36] Chor, B., E. Kushilevitz, O. Goldreich und M. Sudan: *Private information retrieval*. Journal of the ACM (JACM), 45(6):965–981, 1998.

[37] Choudhury, T. und A. Pentland: *Characterizing social interactions using the sociometer*. In: *Proceedings of the 2004 NAACOS Conference*, S. 1–6, 2004.

[38] Chow, C.-Y., M. F. Mokbel und X. Liu: *Spatial cloaking for anonymous location-based services in mobile peer-to-peer environments*. GeoInformatica, 15(2):351–380, 2011.

[39] Constine, J.: *Facebook Launches Nearby Friends With Opt-In Real-Time Location Sharing To Help You Meet Up*. `http://techcrunch.com/2014/04/17/facebook-nearby-friends/`, 2016. letzer Abruf: 17.01.2016.

[40] Corner, M. D. und B. D. Noble: *Zero-interaction authentication*. In: *Proceedings of the 8th annual international conference on Mobile computing and networking (MobiCom 2002)*, S. 1–11. ACM, 2002.

[41] Cranshaw, J., E. Toch, J. Hong, A. Kittur und N. Sadeh: *Bridging the gap between physical location and online social networks*. In: *Proceedings of the 12th ACM international conference on Ubiquitous computing (UbiComp 2010)*, S. 119–128. ACM, 2010.

[42] Diesner, J., T. L. Frantz und K. M. Carley: *Communication networks from the Enron email corpus "It's always about the people. Enron is no different"*. Computational & Mathematical Organization Theory, 11(3):201–228, 2005.

[43] DiNucci, D.: *Fragmented Future*. `http://darcyd.com/fragmented_future.pdf`, 1999. letzer Abruf: 17.01.2016.

[44] Do, T. M. T. und D. Gatica-Perez: *GroupUs: Smartphone Proximity Data and Human Interaction Type Mining*. In: *Proceedings of the 15th Annual International Symposium on Wearable Computers (ISWC 2011)*, S. 21–28. IEEE, 2011.

[45] Duckham, M. und L. Kulik: *A Formal Model of Obfuscation and Negotiation for Location Privacy*. In: *Proceedings of the 3rd International Conference on Pervasive Computing (PERVASIVE 2005)*, Bd. 3468 d. Reihe *Lecture Notes in Computer Science*, S. 152–170. Springer, 2005.

[46] Duckham, M., K. Mason, J. Stell und M. Worboys: *A formal approach to imperfection in geographic information*. Computers, Environment and Urban Systems, 25(1):89–103, 2001.

[47] Ermes, M., J. Parkka, J. Mantyjarvi und I. Korhonen: *Detection of Daily Activities and Sports With Wearable Sensors in Controlled and Uncontrolled Conditions*. IEEE Transactions on Information Technology in Biomedicine, 12(1):20–26, 2008.

[48] Evans, C.: *Notes on the opensurf library*. Techn. Ber. CSTR-09-001, University of Bristol, 2009.

[49] Facebook: *Facebook*. `http://www.facebook.com/`, 2016. letzer Abruf: 07.01.2016.

[50] Facebook: *Nearby Friends*. `https://www.facebook.com/help/629537553762715/`, 2016. letzer Abruf: 17.01.2016.

[51] Fagin, R., M. Naor und P. Winkler: *Comparing information without leaking it*. Communications of the ACM, 39(5):77–85, 1996.

[52] Fitbit, Inc.: *fitbit*. `https://www.fitbit.com/de`, 2016. letzer Abruf: 17.01.2016.

[53] Forsyth, D.: *Group dynamics*. Cengage Learning, 5 Aufl., 2009.

[54] Foursquare: *Foursquare*. `https://foursquare.com/`, 2016. letzer Abruf: 17.01.2016.

[55] Freedman, M. J., K. Nissim und B. Pinkas: *Efficient Private Matching and Set Intersection*. In: *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT 2004)*, Bd. 3027 d. Reihe *Lecture Notes in Computer Science*, S. 1–19. Springer, 2004.

[56] Freudiger, J., M. Raya, M. Félegyházi, P. Papadimitratos et al.: *Mix-zones for location privacy in vehicular networks*. In: *Proceedings of the 2007 ACM Workshop on Wireless Networking for Intelligent Transportation Systems (WiN-ITS 2007)*, S. 1–7. ACM, 2007.

[57] Freudiger, J., R. Shokri und J.-P. Hubaux: *On the Optimal Placement of Mix Zones*. In: *Privacy enhancing technologies*, Bd. 5672 d. Reihe *Lecture Notes in Computer Science*, S. 216–234. Springer, 2009.

[58] Freund, Y. und R. E. Schapire: *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting*. Journal of Computer and System Sciences, 55(1):119–139, 1997.

[59] Fung, E., G. Kellaris und D. Papadias: *Combining Differential Privacy and PIR for Efficient Strong Location Privacy*. In: *Proceedings of the 14th International Symposium on Advances in Spatial and Temporal Databases (SSTD 2015)*, Bd. 9239 d. Reihe *Lecture Notes in Computer Science*, S. 295–312. Springer, 2015.

[60] Gasarch, W.: *A survey on private information retrieval*. In: *Bulletin of the EATCS*, Bd. 82, S. 72–107. EATCS, 2004.

[61] Gauglitz, S., T. Höllerer und M. Turk: *Evaluation of Interest Point Detectors and Feature Descriptors for Visual Tracking*. International Journal of Computer Vision, 94(3):335–360, 2011.

[62] Ge, W., R. T. Collins und B. Ruback: *Automatically detecting the small group structure of a crowd*. In: *Proceedings of the 2009 Workshop on Applications of Computer Vision (WACV 2009)*, S. 1–8. IEEE, 2009.

[63] Gedik, B. und L. Liu: *Location Privacy in Mobile Systems: A Personalized Anonymization Model*. In: *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS 2005)*, S. 620–629. IEEE, 2005.

[64] Gedik, B. und L. Liu: *Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms*. IEEE Transactions on Mobile Computing, 7(1):1–18, 2008.

[65] Ghinita, G., P. Kalnis, A. Khoshgozaran, C. Shahabi und K.-L. Tan: *Private queries in location based services: anonymizers are not necessary*. In: *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, S. 121–132. ACM, 2008.

[66] Ghinita, G., P. Kalnis und S. Skiadopoulos: *MobiHide: A Mobilea Peer-to-Peer System for Anonymous Location-Based Queries*. In: *Proceedings of the 10th International Symposium on Advances in Spatial and Temporal Databases (SSTD 2007)*, Bd. 4605 d. Reihe *Lecture Notes in Computer Science*, S. 221–238. Springer, 2007.

[67] Ghinita, G., P. Kalnis und S. Skiadopoulos: *PRIVE: anonymous location-based queries in distributed mobile systems*. In: *Proceedings of*

the 16th international conference on World Wide Web, S. 371–380. ACM, 2007.

[68] Goldwasser, S., S. Micali und R. L. Rivest: *A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks*. SIAM Journal on Computing, 17(2):281–308, 1988.

[69] Google: *ActivityRecognition*. `https://developers.google.com/android/reference/com/google/android/gms/location/ActivityRecognition`, 2016. letzer Abruf: 10.01.2016.

[70] Google: *Android*. `https://www.android.com/`, 2016. letzer Abruf: 09.01.2016.

[71] Google: *Glass*. `http://www.google.de/glass/start/`, 2016. letzer Abruf: 17.01.2016.

[72] Google: *Google Plus*. `https://plus.google.com/`, 2016. letzer Abruf: 07.01.2016.

[73] Google: *Nexus4*. `https://store.google.com/product/nexus_4_16gb`, 2016. letzer Abruf: 09.01.2016.

[74] Google Play Store: *Fake gps - fake location*. `https://play.google.com/store/apps/details?id=com.fakegps.mock`, 2016. letzer Abruf: 07.01.2016.

[75] Google Play Store: *Fake GPS location*. `https://play.google.com/store/apps/details?id=com.lexa.fakegps`, 2016. letzer Abruf: 07.01.2016.

[76] GoPro: *GoPro*. `https://gopro.com/`, 2016. letzer Abruf: 17.01.2016.

[77] Gordon, D., J.-H. Hanne, M. Berchtold, A. A. N. Shirehjini und M. Beigl: *Towards Collaborative Group Activity Recognition Using Mobile Devices*. Mobile Networks and Applications, 18(3):326–340, 2013.

[78] Gordon, D., M. Scholz und M. Beigl: *Group activity recognition using belief propagation for wearable devices*. In: *Proceedings of the 2014 ACM International Symposium on Wearable Computers*, S. 3–10. ACM, 2014.

[79] Gordon, D., M. Wirz, D. Roggen, G. Tröster und M. Beigl: *Group affiliation detection using model divergence for wearable devices*. In: *Proceedings of the 2014 ACM International Symposium on Wearable Computers*, S. 19–26. ACM, 2014.

[80] Gruteser, M. und D. Grunwald: *Anonymous usage of location-based services through spatial and temporal cloaking*. In: *Proceedings of the 1st international conference on Mobile systems, applications and services*, S. 31–42. ACM, 2003.

[81] Gu, T., Z. Wu, L. Wang, X. Tao und J. Lu: *Mining Emerging Patterns for recognizing activities of multiple users in pervasive computing*. In: *Proceedings of the 6th Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services (MobiQuitous 2009)*, S. 1–10. IEEE, 2009.

[82] Gutscher, A.: *Coordinate transformation-a solution for the privacy problem of location based services?*. In: *Proceedings of the 20th International Parallel and Distributed Processing Symposium (IPDPS 2006)*, S. 7–14. IEEE, 2006.

[83] Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann und I. H. Witten: *The WEKA data mining software: an update*. ACM SIGKDD Explorations Newsletter, 11(1):10–18, 2009.

[84] Hall, M. A.: *Correlation-based Feature Subset Selection for Machine Learning*. Doktorarbeit, University of Waikato, Hamilton, New Zealand, 1998.

[85] Happn: *Happn*. `https://www.happn.com/`, 2016. letzer Abruf: 17.01.2016.

[86] Haritaoglu, I. und M. Flickner: *Detection and tracking of shopping groups in stores*. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, Bd. 1, S. 431–438. IEEE, 2001.

[87] Hazay, C. und Y. Lindell: *Efficient Protocols for Set Intersection and Pattern Matching with Security Against Malicious and Covert Adversaries*. In: *Theory of Cryptography*, Bd. 4948 d. Reihe *Lecture Notes in Computer Science*, S. 155–175. Springer, 2008.

[88] Hazay, C. und K. Nissim: *Efficient Set Operations in the Presence of Malicious Adversaries*. Journal of Cryptology, 25(3):383–433, 2012.

[89] Hightower, J.: *SpotON: An indoor 3D location sensing technology based on RF signal strength*. Techn. Ber. UW CSE 00-02-02, University of Washington, 2000.

[90] Hu, Y.-C., A. Perrig und D. B. Johnson: *Packet leashes: a defense against wormhole attacks in wireless networks*. In: *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications (INFOCOM 2003)*, Bd. 3, S. 1976–1986. IEEE, 2003.

[91] Hutchinson, L.: *iOS 8 to stymie trackers and marketers with MAC address randomization*. http://arstechnica.com/apple/2014/06/ios8-to-stymie-trackers-and-marketers-with-mac-address-randomization/, Juni 2014. accessed on: 2015-06-25.

[92] Huuskonen, P., J. Mäntyjärvi und V. Könönen: *Handbook of Ambient Intelligence and Smart Environments*, Kap. Collaborative Context Recognition for Mobile Devices, S. 257–280. Springer US, 2010.

[93] Huynh, T. und B. Schiele: *Analyzing features for activity recognition*. In: *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies (sOc-EUSAI 2005)*, S. 159–163. ACM, 2005.

[94] IEEE Computer Society: *IEEE Std 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, June 2007.

[95] Jarecki, S. und X. Liu: *Efficient Oblivious Pseudorandom Function with Applications to Adaptive OT and Secure Computation of Set Intersection*. In: *Proceedings of the 6th Theory of Cryptography Conference (TCC 2009)*, Bd. 5444 d. Reihe *Lecture Notes in Computer Science*, S. 577–594. Springer, 2009.

[96] Jones, Q., S. A. Grandhi, L. Terveen und S. Whittaker: *People-to-People-to-Geographical-Places: The P3 Framework for Location-Based Community Systems*. Computer Supported Cooperative Work (CSCW), 13(3-4):249–282, 2004.

[97] Karlsson, N., E. Di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian und M. E. Munich: *The vSLAM Algorithm for Robust Localization and Mapping*. In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA 2005)*, S. 24–29. IEEE, 2005.

[98] Kelley, P. G., R. Brewer, Y. Mayer, L. F. Cranor und N. Sadeh: *An Investigation into Facebook Friend Grouping*. In: *Proceedings of the 13th IFIP TC 13 International Conference on Human-Computer Interaction (INTERACT 2011)*, Bd. 6948 d. Reihe *Lecture Notes in Computer Science*, S. 216–233. Springer, 2011.

[99] Khan, A. M., Y.-K. Lee, S. Lee und T.-S. Kim: *Human Activity Recognition via an Accelerometer-Enabled-Smartphone Using Kernel Discriminant Analysis*. In: *Proceedings of the 5th International Conference on Future Information Technology (FutureTech 2010)*, S. 1–6. IEEE, 2010.

[100] Khan, R., S. Zawoad, M. M. Haque und R. Hasan: *'Who, When, and Where?'Location Proof Assertion for Mobile Devices*. In: *Proceedings of the 28th Annual IFIP WG 11.3 Working Conference on Data and Applications Security and Privacy (DBSec 2014)*, S. 146–162. Springer, 2014.

[101] Kido, H., Y. Yanagisawa und T. Satoh: *An anonymous communication technique using dummies for location-based services*. In: *Proceedings of the 2005 International Conference on Pervasive Services (ICPS 2005)*, S. 88–97. IEEE, 2005.

[102] Kjærgaard, M. B., M. Wirz, D. Roggen und G. Tröster: *Detecting pedestrian flocks by fusion of multi-modal sensors in mobile phones*. In: *Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp 2012)*, S. 240–249. ACM, 2012.

[103] Klette, M.: *Proximitäts-Erkennung anhand optischer Merkmale mit Hilfe von SURF*. Diplomarbeit, Ludwig-Maximilians-Universität München, 2014.

[104] Kortuem, G. und Z. Segall: *Wearable Communities: Augmenting Social Networks with Wearable Computers*. IEEE Pervasive Computing, 2(1):71–78, 2003.

[105] Krumm, J.: *Inference Attacks on Location Tracks*. In: *Proceedings of the 5th International Conference on Pervasive Computing (PERVASIVE 2007)*, Bd. 4480 d. Reihe *Lecture Notes in Computer Science*, S. 127–143. Springer, 2007.

[106] Krumm, J.: *Realistic Driving Trips For Location Privacy*. In: *Proceedings of the 7th International Conference on Pervasive Computing (Pervasive 2009)*, Bd. 5538 d. Reihe *Lecture Notes in Computer Science*, S. 25–41. Springer, 2009.

[107] Küpper, A. und G. Treu: *Efficient proximity and separation detection among mobile targets for supporting location-based community services*. ACM SIGMOBILE Mobile Computing and Communications Review, 10(3):1–12, 2006.

[108] Kushilevitz, E. und R. Ostrovsky: *Replication is not needed: Single database, computationally-private information retrieval*. In: *Proceedings of the 54th IEEE Annual Symposium on Foundations of Computer Science (FOCS 1997)*, S. 364. IEEE, 1997.

[109] LaMarca, A., Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter et al.: *Place Lab: Device Positioning Using Radio Beacons in the Wild*.

In: *Proceedings of the 3rd International Conference on Pervasive Computing (PERVASIVE 2005)*, Bd. 3468 d. Reihe *Lecture Notes in Computer Science*, S. 116–133. Springer, 2005.

[110] Lampson, B. W.: *Protection*. ACM SIGOPS Operating Systems Review, 8(1):18–24, 1974.

[111] Lara, O. D. und M. A. Labrador: *A Survey on Human Activity Recognition using Wearable Sensors*. IEEE Communications Surveys & Tutorials, 15(3):1192–1209, 2013.

[112] Lee, S., M. Kim, S. Kang, K. Lee und I. Jung: *Smart scanning for mobile devices in WLANs*. In: *Proceedings of the 2012 IEEE International Conference on Communications (ICC 2012)*, S. 4960–4964. IEEE, 2012.

[113] Lee, S.-W. und K. Mase: *Activity and location recognition using wearable sensors*. IEEE Pervasive Computing, 1(3):24–32, 2002.

[114] Leonhardt, U. und J. Magee: *Security considerations for a distributed location service*. Journal of Network and Systems Management, 6(1):51–70, 1998.

[115] Leutenegger, S., M. Chli und R. Y. Siegwart: *BRISK: Binary Robust invariant scalable keypoints*. In: *Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV 2011)*, S. 2548–2555. IEEE, 2011.

[116] LG: *LG*. `http://www.lg.com/de`, 2016. letzer Abruf: 09.01.2016.

[117] Li, N., T. Li und S. Venkatasubramanian: *t-Closeness: Privacy Beyond k-Anonymity and l-Diversity*. In: *Proceedings of the 23rd IEEE International Conference on Data Engineering (ICDE 2007)*, S. 106–115. IEEE, 2007.

[118] Lin, Z., D. Foo Kune und N. Hopper: *Efficient Private Proximity Testing with GSM Location Sketches*. In: *Proceedings of the 16th International Conference on Financial Cryptography and Data Security (FC 2012)*, Bd. 7397 d. Reihe *Lecture Notes in Computer Science*, S. 73–88. Springer, 2012.

[119] Lipmaa, H.: *Verifiable Homomorphic Oblivious Transfer and Private Equality Test*. In: *Proceedings of the 9th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2003)*, Bd. 2894 d. Reihe *Lecture Notes in Computer Science*, S. 416–433. Springer, 2003.

[120] Lowe, D. G.: *Object recognition from local scale-invariant features*. In: *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, Bd. 2, S. 1150–1157. IEEE, 1999.

[121] Lowe, D. G.: *Distinctive Image Features from Scale-Invariant Keypoints*. International Journal of Computer Vision, 60(2):91–110, 2004.

[122] Lunden, I.: *6.1B Smartphone Users Globally By 2020, Overtaking Basic Fixed Phone Subscriptions*. `http://techcrunch.com/2015/06/02/6-1b-smartphone-users-globally-by-2020-overtaking-basic-fixed-phone-subscriptions`, 2015. letzer Abruf: 17.01.2016.

[123] Luo, W. und U. Hengartner: *Veriplace: a privacy-aware location proof architecture*. In: *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, S. 23–32. ACM, 2010.

[124] Machanavajjhala, A., D. Kifer, J. Gehrke und M. Venkitasubramaniam: *l-Diversity: Privacy Beyond k-Anonymity*. ACM Transactions on Knowledge Discovery from Data (TKDD), 1(1):1–53, 2007.

[125] Maier, M. und F. Dorfmeister: *Fine-Grained Activity Recognition of Pedestrians Travelling by Subway*. In: *Proceeedings of the 5th International Conference on Mobile Computing, Applications and Services (MobiCASE 2013)*, Bd. 130 d. Reihe *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, S. 122–139. EAI, Springer, 2013.

[126] Maier, M., C. Marouane, M. Klette, F. Dorfmeister, P. Marcus und C. Linnhoff-Popien: *SURFtogether: Towards Context Proximity Detection Using Visual Features*. In: *Proceeedings of the 3rd International Conference on Context-Aware Systems and Applications (ICCASA 2014)*, S. 86–91. EAI, ACM, 2014.

[127] Maier, M., C. Marouane und C. Linnhoff-Popien: *Vis-a-Vis: Offline-Capable Management of Virtual Trust Structures Based on Real-Life Interactions*. International Journal On Advances in Life Sciences, 6(1 and 2):1–10, 2014.

[128] Maier, M., C. Marouane, C. Linnhoff-Popien, B. Rott und S. A. Verclas: *Vis-a-Vis Verification: Social Network Identity Management Through Real World Interactions*. In: *Proceeedings of the 3rd International Conference on Social Eco-Informatics (SOTICS 2013)*, S. 39–44. Thinkmind, 2013.

[129] Maier, M., L. Schauer und F. Dorfmeister: *ProbeTags: Privacy-Preserving Proximity Detection Using Wi-Fi Management Frames*. In:

Proceeedings of the 11th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2015), S. 756–763. IEEE, 2015.

[130] Marin-Perianu, R., M. Marin-Perianu, P. Havinga und H. Scholten: *Movement-Based Group Awareness with Wireless Sensor Networks*. In: *Proceedings of the 5th International Conference on Pervasive Computing (PERVASIVE 2007)*, Bd. 4480 d. Reihe *Lecture Notes in Computer Science*, S. 298–315. Springer, 2007.

[131] Mascetti, S., C. Bettini, D. Freni, X. S. Wang und S. Jajodia: *Privacy-Aware Proximity Based Services*. In: *Proceedings of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware (MDM 2009)*, S. 31–40. IEEE, 2009.

[132] Mascetti, S., C. Bettini, X. S. Wang, D. Freni und S. Jajodia: *ProvidentHider: An Algorithm to Preserve Historical k-Anonymity in LBS*. In: *Proceedings of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware (MDM 2009)*, S. 172–181. IEEE, 2009.

[133] Mathur, S., R. Miller, A. Varshavsky, W. Trappe und N. Mandayam: *ProxiMate: Proximity-based Secure Pairing Using Ambient Wireless Signals*. In: *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys 2011)*, S. 211–224. ACM, 2011.

[134] Michalevsky, Y., D. Boneh und G. Nakibly: *Gyrophone: Recognizing Speech from Gyroscope Signals*. In: *Proceedings of the 23rd USENIX Security Symposium (SEC 2014)*, S. 1053–1067. USENIX, 2014.

[135] Mikolajczyk, K. und C. Schmid: *A performance evaluation of local descriptors*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(10):1615–1630, 2005.

[136] Miluzzo, E., N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng und A. T. Campbell: *Sensing meets mobile social networks: the design, implementation and evaluation of the CenceMe application*. In: *Proceedings of the 6th ACM conference on Embedded network sensor systems (SenSys 2008)*, S. 337–350. ACM, 2008.

[137] Mokbel, M. F., C.-Y. Chow und W. G. Aref: *The new Casper: query processing for location services without compromising privacy*. In: *Proceedings of the 32nd international conference on Very large data bases (VLDB 2006)*, S. 763–774. ACM, 2006.

[138] Moussaïd, M., N. Perozo, S. Garnier, D. Helbing und G. Theraulaz: *The walking behaviour of pedestrian social groups and its impact on crowd dynamics.*. PloS one, 5(4), 2009. http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0010047.

[139] Mozilla: *Location Service*. https://location.services.mozilla.com/, 2016. letzer Abruf: 23.01.2016.

[140] Muñoz-Organero, M., P. J. Muñoz-Merino und C. Delgado Kloos: *Using Bluetooth to Implement a Pervasive Indoor Positioning System with Minimal Requirements at the Application Level*. Mobile Information Systems, 8(1):73–82, 2012.

[141] Müller, U.: *Fußgänger-Kontexterkennung in der U-Bahn*. Diplomarbeit, Ludwig-Maximilians-Universität München, 2013.

[142] Münchner Merkur: *Fahrgastrekord: MVG warnt vor Kollaps*. http://www.merkur.de/lokales/muenchen/stadt-muenchen/fahrgastrekord-warnt-kollaps-5173863.html, 2016. letzer Abruf: 10.01.2016.

[143] Nam, Y., S. Rho und C. Lee: *Physical activity recognition using multiple sensors embedded in a wearable device*. ACM Transactions on Embedded Computing Systems (TECS), 12(2):26:1–26:14, 2013.

[144] Narayanan, A., N. Thiagarajan, M. Lakhani, M. Hamburg und D. Boneh: *Location Privacy via Private Proximity Testing*. In: *Proceedings of the 18th Annual Network & Distributed System Security Symposium (NDSS 2011)*, S. 1–16. ISOC, 2011.

[145] Navarro, G.: *A guided tour to approximate string matching*. ACM computing surveys (CSUR), 33(1):31–88, 2001.

[146] Nielsen, J., J. Pagter und M. Stausholm: *Location Privacy via Actively Secure Private Proximity Testing*. In: *Proceedings of the 2012 IEEE Pervasive Computing and Communications Workshops (PERCOM Workshops 2012)*, S. 381–386. IEEE, 2012.

[147] OpenWLANMap: *OpenWLANMap*. http://openwlanmap.org/, 2016. letzer Abruf: 23.01.2016.

[148] O'Reilly, T.: *What Is Web 2.0*. http://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html, 2005. letzer Abruf: 17.01.2016.

[149] Palanisamy, B. und L. Liu: *MobiMix: Protecting location privacy with mix-zones over road networks*. In: *Proceedings of the 27th International Conference on Data Engineering (ICDE 2011)*, S. 494–505. IEEE, 2011.

[150] Palazzi, C. E.: *Buddy-Finder: A proposal for a novel entertainment application for GSM*. In: *Proceedings of the 2004 IEEE Global Telecommunications Conference Workshops (GlobeCom Workshops 2004)*, S. 540–543. IEEE, 2004.

[151] Pietiläinen, A.-K., E. Oliver, J. LeBrun, G. Varghese und C. Diot: *MobiClique: middleware for mobile social networking*. In: *Proceedings of the 2nd ACM workshop on Online social networks (WOSN 2009)*, S. 49–54. ACM, 2009.

[152] Poppe, R.: *A survey on vision-based human action recognition*. Image and Vision Computing, 28(6):976–990, 2010.

[153] Qiu, D., D. Boneh, S. Lo und P. Enge: *Robust location tag generation from noisy location data for security applications*. In: *The Institute of Navigation International Technical Meeting*. Citeseer, 2009.

[154] Ravi, N., N. Dandekar, P. Mysore und M. L. Littman: *Activity recognition from accelerometer data*. In: *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI 2005)*, S. 1541–1546. AAAI, 2005.

[155] Reddy, S., J. Burke, D. Estrin, M. Hansen und M. Srivastava: *Determining transportation mode on mobile phones*. In: *Proceedings of the 12th IEEE International Symposium on Wearable Computers (ISWC 2008)*, S. 25–28. IEEE, 2008.

[156] Robson, S.: *Six of the world's seven billion people have mobile phones... but only 4.5billion have a toilet says UN report*. `http://www.dailymail.co.uk/news/article-2297508/Six-world-s-seven-billion-people-mobile-phones--4-5billion-toilet-says-UN-report.html`, 2013. letzer Abruf: 17.01.2016.

[157] Roggen, D., M. Wirz, G. Tröster und D. Helbing: *Recognition of Crowd Behavior from Mobile Sensors with Pattern Analysis and Graph Clustering Methods*. Networks and Heterogenous Media, 6(3):521–544, 2011.

[158] Saroiu, S. und A. Wolman: *Enabling new mobile applications with location proofs*. In: *Proceedings of the 10th workshop on Mobile Computing Systems and Applications (HotMobile 2009)*, S. 3–8. ACM, 2009.

[159] Sastry, N., U. Shankar und D. Wagner: *Secure verification of location claims*. In: *Proceedings of the 2nd ACM workshop on Wireless security (WiSe 2003)*, S. 1–10. ACM, 2003.

[160] Schauer, L., M. Werner und P. Marcus: *Estimating crowd densities and pedestrian flows using wi-fi and bluetooth*. In: *Proceedings of the 11th*

International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, S. 171–177. ICST, 2014.

[161] Schilit, B., N. Adams und R. Want: *Context-Aware Computing Applications*. In: *Proceedings of the 1st Workshop on Mobile Computing Systems and Applications (WMCSA 1994)*, S. 85–90. IEEE, 1994.

[162] Schmid, C., R. Mohr und C. Bauckhage: *Evaluation of Interest Point Detectors*. International Journal of Computer Vision, 37(2):151–172, 2000.

[163] Schroth, G., R. Huitl, D. Chen, M. Abu-Alqumsan, A. Al-Nuaimi und E. Steinbach: *Mobile Visual Location Recognition*. IEEE Signal Processing Magazine, 28(4):77–89, 2011.

[164] Schuster, D., A. Rosi, M. Mamei, T. Springer, M. Endler und F. Zambonelli: *Pervasive social context: Taxonomy and survey*. ACM Transactions on Intelligent Systems and Technology (TIST), 4(3):46:1–46:22, 2013.

[165] Shankar, P., V. Ganapathy und L. Iftode: *Privately querying location-based services with SybilQuery*. In: *Proceedings of the 11th international conference on Ubiquitous computing (UbiComp 2009)*, S. 31–40. ACM, 2009.

[166] Shi, J. und C. Tomasi: *Good features to track*. In: *Proceedings of the 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 1994)*, S. 593–600. IEEE, 1994.

[167] Šikšnys, L., J. R. Thomsen, S. Šaltenis, M. L. Yiu und O. Andersen: *A Location Privacy Aware Friend Locator*. In: *Proceedings of the 11th International Symposium on Advances in Spatial and Temporal Databases (SSTD 2009)*, S. 405–410. Springer, 2009.

[168] Sion, R. und B. Carbunar: *On the Practicality of Private Information Retrieval*. In: *Proceedings of the 14th Network and Distributed Systems Security Symposium (NDSS 2007)*, S. 1–10. ISOC, 2007.

[169] Snekkenes, E.: *Concepts for personal location privacy policies*. In: *Proceedings of the 3rd ACM conference on Electronic Commerce (EC 2001)*, S. 48–57. ACM, 2001.

[170] Solanas, A., F. Sebé und J. Domingo-Ferrer: *Micro-aggregation-based heuristics for p-sensitive k-anonymity: one step beyond*. In: *Proceedings of the 2008 international workshop on Privacy and anonymity in information society (PAIS 2008)*, S. 61–69. ACM, 2008.

[171] Statista: *Das Smartphone (fast) immer im Blick*. http://de.statista.com/infografik/4192/nutzungsmuster-von-mobiltelefonnutzern/, 2016. letzer Abruf: 07.01.2016.

[172] Steggles, P. und S. Gschwind: *The Ubisense smart space platform*.

[173] Stenneth, L., O. Wolfson, P. S. Yu und B. Xu: *Transportation mode detection using mobile phones and GIS information*. In: *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS 2011)*, S. 54–63. ACM, 2011.

[174] Stonerworx: *OpenSURF-Java*. https://github.com/stonerworx/opensurf-java, 2016. letzer Abruf: 28.02.2016.

[175] Sun, L., D. Zhang, B. Li, B. Guo und S. Li: *Activity Recognition on an Accelerometer Embedded Mobile Phone with Varying Positions and Orientations*. In: Yu, Z., R. Liscano, G. Chen, D. Zhang und X. Zhou (Hrsg.): *Ubiquitous Intelligence and Computing*, Bd. 6406 d. Reihe *Lecture Notes in Computer Science*, S. 548–562. Springer Berlin Heidelberg, 2010.

[176] Sun, L., D. Zhang, B. Li, B. Guo und S. Li: *Activity Recognition on an Accelerometer Embedded Mobile Phone with Varying Positions and Orientations*. In: *Proceedings of the 7th International Conference on Ubiquitous Intelligence and Computing (UIC 2010)*, S. 548–562. Springer, 2010.

[177] Sweeney, L.: *k-anonymity: A model for protecting privacy*. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(05):557–570, 2002.

[178] Synty Studios: *Facebook Page*. https://www.facebook.com/syntystudios, 2016. letzer Abruf: 28.02.2016.

[179] Talukder, N. und S. I. Ahamed: *Preventing multi-query attack in location-based services*. In: *Proceedings of the 3rd ACM conference on Wireless network security (WiSec 2010)*, S. 25–36. ACM, 2010.

[180] Tinder, Inc.: *Tinder*. https://www.gotinder.com/, 2016. letzer Abruf: 10.01.2016.

[181] Torres-Sospedra, J., R. Montoliu, S. Trilles, Ó. Belmonte und J. Huerta: *Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems*. Expert Systems with Applications, 42(23):9263–9278, 2015.

[182] Treu, G.: *Efficient Proximity Detection for Location Based Services*. In: *Proceedings of the Joint 2nd Workshop on Positioning, Navigation and*

Communication 2005 (WPNC05) and 1st Ultra-Wideband Expert Talk (UET05). Citeseer, 2005.

[183] Treu, G., A. Küpper und P. Ruppel: *Anonymization in proactive location based community services*. 2005.

[184] Twitter, Inc.: *Twitter*. `https://twitter.com/`, 2016. letzer Abruf: 07.01.2016.

[185] Wang, S., C. Chen und J. Ma: *Accelerometer based transportation mode recognition on mobile phones*. In: *Proceedings of the Asia-Pacific Conference on Wearable Computing Systems (APWCS 2010)*, S. 44–46. IEEE, 2010.

[186] Wang, Y., X. Yang, Y. Zhao, Y. Liu und L. Cuthbert: *Bluetooth positioning using RSSI and triangulation methods*. In: *Proceedings of the Consumer Communications and Networking Conference (CCNC 2013)*, S. 837–842. IEEE, 2013.

[187] Want, R., A. Hopper, V. Falcao und J. Gibbons: *The active badge location system*. ACM Transactions on Information Systems (TOIS), 10(1):91–102, 1992.

[188] Weiser, M.: *Hot topics-ubiquitous computing*. IEEE Computer, 26(10):71–72, 1993.

[189] Weiser, M.: *Some computer science issues in ubiquitous computing*. Communications of the ACM, 36(7):75–84, 1993.

[190] Weiser, M.: *Ubiquitous computing*. IEEE Computer, 26(10):71–72, 1993.

[191] Wenning, R. und M. Schunter: *The Platform for Privacy Preferences 1.1 (P3P1.1) Specification*. Techn. Ber., W3C, 2006. `http://www.w3.org/TR/2006/NOTE-P3P11-20061113/`.

[192] Weppner, J. und P. Lukowicz: *Collaborative crowd density estimation with mobile phones*. In: *Proceedings of the 2nd International Workshop on Sensing Applications on Mobile Phones (PhoneSense 2011)*, 2011.

[193] Werner, M., F. Dorfmeister und M. Schönfeld: *AMBIENCE: A Context-Centric Online Social Network*. In: *Proceedings of the workshop on Positioning, Navigation and Communications (WPNC 2015)*, 2015. to appear.

[194] Werner, M., M. Kessel und C. Marouane: *Indoor positioning using smartphone camera*. In: *Proceedings of the 2011 International Conference on Indoor Positioning and Indoor Navigation (IPIN 2011)*, S. 1–6, 2011.

[195] Wernke, M., P. Skvortsov, F. Dürr und K. Rothermel: *A classification of location privacy attacks and approaches*. Personal and Ubiquitous Computing, 18(1):163–175, 2014.

[196] Wikipedia: *Haversine Formula*. `https://en.wikipedia.org/w/index.php?title=Haversine_formula&oldid=697936784`, 2016. letzer Abruf: 17.01.2016.

[197] Wikipedia: *iPhone*. `https://en.wikipedia.org/w/index.php?title=IPhone&oldid=698818825`, 2016. letzer Abruf: 10.01.2016.

[198] World Wide Web Consortium: *World Wide Web Consortium*. `http://www.w3.org/`, 2016. letzer Abruf: 08.01.2016.

[199] Yatani, K. und K. N. Truong: *BodyScope: a wearable acoustic sensor for activity recognition*. In: *Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp 2012)*, S. 341–350. ACM, 2012.

[200] Zhang, C. und Y. Huang: *Cloaking locations for anonymous location based services: a hybrid approach*. GeoInformatica, 13(2):159–182, 2009.

[201] Zhang, M. und A. A. Sawchuk: *A feature selection-based framework for human activity recognition using wearable multimodal sensors*. In: *Proceedings of the 6th International Conference on Body Area Networks (BodyNets 2011)*, S. 92–98. ICST, 2011.

[202] Zhang, Z. und S. Poslad: *Fine-Grained Transportation Mode Recognition Using Mobile Phones and Foot Force Sensors*. In: *Proceedings of the 9th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (Mobiquitous 2012)*, Bd. 120 d. Reihe *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, S. 103–114. Springer, 2012.

[203] Zheng, Y., M. Li, W. Lou und Y. Hou: *SHARP: Private Proximity Test and Secure Handshake with Cheat-Proof Location Tags*. In: *Proceedings of the 17th European Symposium on Research in Computer Security (ESORICS 2012)*, Bd. 7459 d. Reihe *Lecture Notes in Computer Science*, S. 361–378. Springer, 2012.

[204] Zheng, Y., Q. Li, Y. Chen, X. Xie und W.-Y. Ma: *Understanding mobility based on GPS data*. In: *Proceedings of the 10th international conference on Ubiquitous computing (UbiComp 2008)*, S. 312–321. ACM, 2008.