In ancient days, localisation meant navigation - an art of finding the way from one place to another. Tremendous advancement in the science of navigation dates back to the sixteenth century, when instruments like compasses, sextants and the first ever clock to keep the time exactly were devised. Since then localisation has been explored for several decades as a classical problem in many disciplines. Now we are in the era of ubiquitous computing - a computing paradigm of the future, where various computing elements will be seamlessly integrated into the environment and will be invisible to the user. Knowing the location of an object is an important cornerstone and a fertile research area in ubiquitous computing.

Fundamental to any location system are the algorithms used to estimate location. No location system is error-free and suited for all situations, for example, pure inertial sensors suffer from drift, ultrasound sensors require clear line of sight and magnetic sensors are affected by ferromagnetic and conductive materials in the environment. Thus, we rationalise "multimodal localisation" as one of the promising ways for improving accuracy.

In this thesis, we explore localisation algorithms that use multiple sensing modalities obtained from some of the most commonly used technologies such as WLAN, UWB, ultrasound and inertial sensors. In every instance, we have illustrated the benefits of combining measurements from multiple modalities. The specific contributions include, motion detection algorithms and technology independent localisation algorithms that have the ability to fuse readings across different sensing technologies and incorporate motion models to improve accuracy significantly. Another important aspect of the work presented in this thesis is the characterisation of the raw measurement errors of the individual modalities. In all cases, we perform a rigorous evaluation of the presented algorithms by using measurements collected from real deployments.

ISBN: 978-90-365-2890-0

MULTIMODAL LOCALISATION
Analysis, Algorithms and Experimental Evaluation
Kavitha Muthukrishnan

MULTIMODAL LOCALISATION

Analysis, Algorithms and Experimental Evaluation

Kavitha Muthukrishnan

MULTIMODAL LOCALISATION

– Analysis, Algorithms and Experimental Evaluation –

Kavitha Muthukrishnan

Composition of the Graduation Committee:

prof.dr. P.J.M. Havinga	University of Twente, The Netherlands (Promotor)
prof.dr.ir. G.J.M. Smit	University of Twente, The Netherlands
prof.dr. S.J. Mullender	University of Twente, The Netherlands
prof.dr. H.W. Gellersen	Lancaster University, UK
prof.dr. K.G. Langendoen	Technical University Delft, The Netherlands
dr. M. Hazas	Lancaster University, UK
prof.dr. I.G.M.M. Niemegeers	Technical University Delft, The Netherlands

This research was conducted within the BSIK Smart Surroundings project.

Pervasive Systems Research Group
Faculty of Electrical Engineering,
Mathematics and Informatics
P.O. Box 217 7500 AE
Enschede, The Netherlands.

Series title: CTIT Ph.D.-thesis Series
ISSN: 1381- 3617
CTIT Number: 09-153

This thesis was edited with TexShop and typeset with LaTeX2e.

Keywords: localisation and tracking, motion inference, sensor fusion, Kalman filtering, sensor data characterisation, WLAN, ultra-wideband, dead reckoning, ultrasound, algorithms, experimental evaluation.

Cover Design: Kiran.K.Thumma; Illustration of Hansel and Gretel's breadcrumbs.

Printed by : Wöhrmann Print Service, Zutphen, The Netherlands.
ISBN: 978-90-365-2890-0
DOI: 10.3990/1.9789036528900

MULTIMODAL LOCALISATION

— Analysis, Algorithms and Experimental Evaluation —

DISSERTATION

to obtain
the degree of doctor at the University of Twente,
on the authority of the rector magnificus,
prof.dr. H.Brinksma,
on account of the decision of the graduation committee,
to be publicly defended
on Friday, 4 September 2009 at 15:00 hrs

by

Kavitha Muthukrishnan

born on 15 March 1979
in Chennai, India

This dissertation is approved by:
prof.dr. P.J.M. Havinga (Promotor)

To my parents and my husband, Kiran

Abstract

The term *localisation* is derived from the word *locale*, which traditionally means a small area or vicinity. In ancient days, localisation meant navigation – an art of finding the way from one place to another. Tremendous advancement in the science of navigation dates back to the sixteenth century, when instruments like compasses, sextants and the first ever clock to keep the time exactly were devised. Advancement in navigation brought ways and means to explore the world, be it for expansion of the territories or for promoting trade and business. Since then localisation has been explored for several decades as a classical problem in many disciplines – including robotics, virtual reality, navigation. Now we are in the era of ubiquitous computing – a term coined by the visionary Mark Weiser in the early 1990s. Weiser sees technology only as a means to an end, which should take a back seat in order to allow user to fully concentrate on the task at hand. Looking from a technological standpoint, today we are surrounded by a wealth of devices enriched with *sensing*, *computing* and *communication* capabilities which are seamlessly integrated in our daily lives. Knowing the *location* of an object is an important cornerstone and fertile research area in ubiquitous computing.

The growing need of location systems underscores the importance of addressing this problem – government initiatives to locate emergency call by cellular network providers and the increasing usage of global positioning systems (GPS) in many commercial applications as in navigation are just a few examples. Since the field is active and vibrant, new services and market players are constantly emerging. Google have just launched a new service called Latitude, which lets smart phone and laptop users share their location with friends and allows those friends to share their locations in return. Latitude uses satellites and cell towers to estimate location. The market for GPS products and services alone is expected to grow to US$ 200 billion by 2015 [167]. Real-time location systems (RTLS) in the transport and logistics sector drive the penetration of several location-based solutions. The number of RTLS suppliers is expected to increase from 50 to 200 by 2013, reflecting a market growth from $145 million in 2008 to $2.7 billion in 2013 [22]. Despite the extraordinary advances in outdoor localisation and navigation, indoor localisation still remains an open challenge.

Fundamental to any location system are the *algorithms* used to estimate location. This thesis focuses on formulation of *localisation algorithms* with the capability of fusing measurements from multiple modalities. We begin by systematically analysing the basic principles of localisation through a review and classification of the state of the art. From our detailed survey, it is evident that no location system is error-free and suited for all situations. For example, pure inertial sensors suffer from drift, ultrasound sensors require clear line of sight and magnetic sensors are affected by ferromagnetic and conductive materials in the environment. Thus, we rationalise *multimodal localisation* as one of the promising ways for improving location accuracy. Apart from improving performance of the location system in limited measurement volumes, fusion of heterogeneous sensing systems will ultimately

allow people to move between places covered by different sensing systems without loss of location knowledge.

We explore localisation algorithms that use multiple sensing modalities to improve accuracy and robustness. To ground our work, we have chosen three specific applications covering both infrastructure-based positioning and ad hoc-based positioning systems. From our taxonomy, we create a blueprint of location technologies that would meet those three application needs.

1. Localisation in office environments to facilitate social networking, as a way to help coordination of people and understand social patterns. We leverage the existing wireless local-area networks (WLAN) infrastructure to sense motion and location with the main motivation of building wide-area location services. Our contributions include– (i) in-depth characterisation of received signal strength (RSSI), (ii) novel algorithms to deduce motion by observing fluctuations in RSSI across all the access points in range, and (iii) performance comparison using real data against common deterministic location algorithms with and without adding motion information.

2. Transport and logistics operation (e.g. in warehouses), motivating the need of fine-grained location information. We use ultra-wideband (UWB) as it copes with harsh indoor environments better than conventional radio technologies. Our contributions include– (i) characterisation of heterogeneous observations (pseudoranges and angles) obtained from two deployments, mimicking real-world (low-overhead) vs. ideal deployment (carefully planned and calibrated), (ii) formulation of algorithms to fuse heterogeneous observations and (iii) a thorough evaluation for both static and dynamic tracking.

3. Emergency response scenarios, motivating the need for ad hoc positioning capabilities. In particular, we use a combination of inertial sensors and ultrasound sensors. The position error in a purely inertial system increases with time and requires correction from external sources. We address this problem by deploying ultrasound sensors as landmarks correcting for the inertial drift. Our contributions include– (i) characterisation of inertial and ultrasound data, (ii) algorithms to support guidance and tracking and (iii) a thorough evaluation from data gathered from real deployments.

While the chosen technologies and applications are not exhaustive, they are representative as they cover a broad spectrum across several dimensions: *accuracy* – fine grained to coarse grained, *coverage* – room-level to wide-area, *dependence* – dense infrastructure to ad hoc, *cost* – expensive to minimal cost. In every instance, we have illustrated the benefits of combining multiple modalities.

In short, our contributions include algorithms for motion detection and technology independent localisation algorithms that have the ability to fuse readings across different sensing technologies and incorporate motion models to improve accuracy significantly. Another important aspect of the work presented in this thesis is the characterisation of the raw measurement errors of the individual modalities. In all cases, we perform a rigorous evaluation of the presented algorithms by using measurements collected from real deployments.

Samenvatting

Lang geleden betekende lokalisatie (plaatsbepaling) hetzelfde als navigatie – de kunst om de weg te vinden van A naar B. Al in de zestiende eeuw werd op het gebied van navigatie enorme wetenschappelijke vooruitgang geboekt, met de ontwikkeling van instrumenten zoals het kompas, de sextant, en de allereerste uurwerken die exact de tijd bijhielden. Deze vooruitgang opende nieuwe mogelijkheden om de wereld te ontdekken, of het nu voor de verovering van nieuwe gebieden was of voor het drijven van handel. Later is lokalisatie gedurende meerdere decennia als een klassiek probleem het onderwerp geworden van studie in vele disciplines, zoals robotica, virtuele realiteit, en navigatie. We zijn nu aangekomen in het tijdperk van *ubiquitous computing* (alomtegenwoordige informatietechnologie) – een term bedacht door visionair Mark Weiser in de vroege jaren 1990. Weiser ziet technologie slechts als een middel om een doel te bereiken. Daarbij zou de technologie ergens in de achtergrond moeten blijven en kan de gebruiker zich op die manier volledig concentreren op de taak die voor hem ligt. Gezien vanuit een technologisch perspectief zijn we vandaag de dag omgeven door een overvloed aan apparaten die kunnen *waarnemen*, *rekenen* en *communiceren*, en die naadloos geïntegreerd zijn in ons dagelijks leven. De *locatie* van een object te kennen vormt een belangrijke hoeksteen van ubiquitous computing en is daardoor een vruchtbaar onderzoeksgebied.

De groeiende behoefte aan lokalisatiesystemen onderstreept hoe belangrijk het is om dit probleem aan te pakken – overheidsinitiatieven voor de lokalisatie van noodoproepen via mobielnetwerkaanbieders en het groeiende gebruik van GPS (global positioning systems) in vele commerciële toepassingen zoals in navigatie zijn slechts enkele voorbeelden. Het is een actief en levendig onderzoeksgebied, dus is er een constant groeiend aanbod van nieuwe diensten en spelers op de markt. Google heeft pas nog een nieuwe dienst gelanceerd met de naam Latitude, die gebruikers van smart phones en laptops hun locatie laat delen met vrienden die op hun beurt de mogelijkheid hebben hun eigen locatie ook kenbaar te maken. Latitude gebruikt satellieten en zendmasten voor mobiele telefonie voor plaatsbepaling. Alleen al de markt voor GPS-producten en -diensten zal naar verwachting groeien naar US$ 200 miljard in 2015 [167]. Real-time lokalisatiesystemen (RTLS) zijn de drijfveer achter het binnendringen van verschillende plaatsgerelateerde oplossingen in de transport- en logistieksector. Het aantal aanbieders van RTLS zal naar verwachting groeien van 50 naar 200 in 2013, wat een marktgroei betekent van $ 145 miljoen in 2008 naar $ 2,7 miljard in 2013 [22]. Ondanks de buitengewone vooruitgang in plaatsbepaling en navigatie in buitenomgevingen, blijft plaatsbepaling in gebouwen nog steeds een uitdaging.

Aan de basis van ieder lokalisatiesysteem staan de algoritmen die gebruikt worden voor de plaatsbepaling. Dit proefschrift concentreert zich op het formuleren van *lokalisatiealgoritmen* die metingen uit verschillende soorten systemen kunnen combineren. Om te beginnen analyseren wij op systematische wijze de grondbeginselen van lokalisatie via een bespreking en een classificatie van de huidige stand van de techniek. Onze overzichtsstudie toont aan dat geen enkel plaatsbepalingssysteem foutvrij en geschikt voor alle situaties is. Zo hebben puur inertiële sensoren last van drift, vereisen ultrasoonsensoren een vrije zichtverbinding, en worden magnetische sensoren beïnvloed door ferromagnetische en geleidende materialen in de omgeving. Derhalve beredeneren we dat *multimodale lokalisatie* een veelbelovende manier is om de lokalisatienauwkeurigheid te verbeteren. Naast verbeterde prestaties van het plaatsbepalingssysteem bij een beperkt aantal metingen, zal het combineren van heterogene waarnemingssystemen mensen uiteindelijk de mogelijkheid bieden tussen locaties te bewegen die gedekt worden door verschillende waarnemingssystemen, zonder verlies van locatiekennis.

We verkennen lokalisatiealgoritmen die gebruik maken van verschillende waarnemingssystemen om de nauwkeurigheid en robuustheid te verbeteren. Als fundament voor ons werk hebben we drie specifieke toepassingen gekozen die samen zowel plaatsbepaling gebaseerd op infrastructuur als ad hoc plaatsbepaling omvatten. Vanuit onze taxonomie creëren we een blauwdruk voor lokalisatietechnologieën die aan de vereisten voor die drie toepassingen voldoen.

1. Lokalisatie in kantooromgevingen om sociale netwerken te faciliteren, als een manier om de coördinatie van mensen te ondersteunen en sociale patronen te begrijpen. We maken doelmatig gebruik van de bestaande WLAN-infrastructuur (wireless local-area networks) voor bewegings- en plaatswaarneming met als belangrijkste motivatie het bouwen van locatiediensten die een groot gebied bestrijken. Onze bijdragen omvatten (i) een diepgaande karakterisatie van RSSI-gegevens (ontvangen signaalsterkte), (ii) nieuwe algoritmen om beweging af te leiden door fluctuaties in RSSI waar te nemen over alle WLAN access points (toegangspunten) binnen bereik, en (iii) een prestatievergelijking tussen veelgebruikte deterministische lokalisatiealgoritmen zowel met als zonder toevoeging van bewegingsinformatie, op basis van meetgegevens uit de praktijk.

2. Transport en logistiek (bijvoorbeeld in opslagloodsen) als motivatie voor de noodzaak van gedetailleerde locatiekennis; onze keus was het gebruik van ultrabreedband (ultrawideband, UWB) omdat dat beter in staat is om te gaan met lastige binnenomgevingen dan conventionele radiotechnologie. Onze bijdragen omvatten (i) een karakterisatie van heterogene observaties (pseudobereiken en hoeken) verkregen uit twee opstellingen, die een praktische tegenover een ideale toepassing nabootsen, (ii) een formulering van algoritmen om heterogene observaties samen te voegen, en (iii) een grondige evaluatie van zowel statische als dynamische tracking (locatievolgtechnieken).

3. Calamiteitenscenario's als motivatie voor de noodzaak om ad hoc positiebepaling te kunnen uitvoeren. In het bijzonder gebruiken we een combinatie van inertiële sensoren en ultrasoonsensoren. The positiefout in een puur inertieel systeem neemt toe met de tijd en vereist extern gestuurde correctie. We pakken dit probleem aan door ultrasoonsensoren te plaatsen als oriëntatiepunten voor de correctie van de inertiële drift. Onze bijdragen omvatten (i) een karakterisatie van inertiële en ultrasone gegevens en (ii) algoritmen om begeleiding en tracking te ondersteunen, en (iii) een grondige evaluatie van gegevens verzameld uit de praktijk.

Hoewel de lijst van gekozen technologieën en toepassingen niet uitputtend is, is deze wel representatief omdat zij een breed spectrum omvat over meerdere dimensies: *nauwkeurigheid* – van fijnmazig tot grofmazig, *dekkingsgraad* – van kamerniveau tot wide-area, *afhankelijkheid* – van infrastructuur met hoge dichtheid tot ad hoc (zonder infrastructuur), *kosten* – van duur tot minimale kosten. Voor ieder geval hebben we de voordelen van het combineren van verschillende soorten systemen geïllustreerd.

Kortom, onze bijdragen omvatten bewegingsdetectiealgoritmen en technologieonafhankelijke lokalisatiealgoritmen die het vermogen hebben metingen uit verschillende waarnemingstechnologieën te combineren en die bewegingsmodellen in zich hebben om de nauwkeurigheid significant te verbeteren. Een ander belangrijk aspect van het werk in dit proefschrift is de karakterisatie van de ruwe meetfouten van de verschillende systemen. In alle gevallen voeren we een nauwgezette evaluatie uit van de gepresenteerde algoritmen door gebruik te maken van metingen die we uit de praktijk hebben verzameld.

Acknowledgements

Looking back, approximately six-and-a-half-years ago when I came to the Netherlands for pursuing my Master's degree, I could never have imagined pursuing a PhD. I have always believed PhD is something which requires lots of hardwork, perseverance, patience, dedication and felt that I lack many of these attributes. Pursuing a Master's degree here, changed my outlook and made me apply for a PhD position with great confidence.

My thanks must begin with my promotor and supervisor, Prof. Paul Havinga for giving me an opportunity to work in this fascinating field of ubiquitous computing. I sincerely appreciate the patience he had in me over these years. There were instances when certain things took longer time to address but, he gave me freedom and encouragement. His support throughout these years and especially, at the time of writing of this thesis were commendable. I am grateful to him for funding my stay at Lancaster and also for providing extensions to my contract, without which my thesis would not have reached this stage. Paul, I am extremely delighted and privileged that you are promoting me as your first student. My highly learned promotor—thanks for all your support.

I am grateful to Gerard for providing such a thorough comments on my thesis. All his comments were extremely valuable in shaping up this thesis.

My stay at Lancaster was the golden period in my Phd. I would like to sincerely thank Prof. Hans Gellersen for inviting me to the Embedded Interactive Systems group at Lancaster and for connecting me with some of the great people to collaborate with. I appreciate the freedom and motivation that he gave me and also for taking time to discuss about my research, apart from the work I was doing at Lancaster.

Many heart felt thanks to Dr. Mike Hazas with whom I have enjoyed working, the last couple of years. I must also thank him for being part of my committee and for providing extremely valuable comments, without which my thesis would not have reached this far. He is a perfectionist, and I have learnt a lot of things from him, starting from how to do be a good researcher and write scientific papers till making spicy salsa! I have always admired the dedication and support he has shown towards his students and was lucky to get a proxy supervision. Apart from our research collaboration, he has been a great friend and it was so very thoughtful of him to have arranged a lovely farewell before I left Lancaster. No doubt that he has been a great source of inspiration to lead an academic path and I sincerely hope that our collaboration would continue in future.

Many thanks to Prof. Koen Langendoen for the comments and suggestions on my thesis. I am extremely delighted and excited about becoming a part of the Embedded Software chair at TUDelft soon after my defence.

I would like to express my gratitude to the members of my promotion committee for taking time to read my dissertation and for assessing the manuscript.

I would like to thank the Ministry of Economic Affairs of the Netherlands for funding my research through the project Smart Surroundings (contract no: 03060). I would also like to acknowledge the support that I received from EU-funded e-Sense project (grant no: 027227) and the EU-funded Relate project (grant no: 013790).

Many thanks to Maria for being my daily supervisor in the initial years of my PhD. She extended her support for many things, be it for the quick arrangement relating to funding or for the supervision. The most striking thing in her is providing an honest feedback. Though at times, receiving honest feedback can be tough, it has helped me to be a better researcher.

Nirvana has been my daily supervisor in the initial years and a well-wisher. Her support was extremely valuable. Together with Maria, she was of great help in arranging funding and for the supervision. I must also thank her for proof reading (many) earlier versions of some of the chapters of my thesis. She has never said "no" to any of my requests.

I am particularly indebted to Berend Jan for his invaluable support. Although we worked together towards the tail end of my PhD, I am extremely fortunate to have worked with him. He has an eye for perfection in what ever he does. He has been there "altijd" for discussing about the FFT's and RSSI analysis. I appreciate his timely-help during the measurements and for the wonderful job of translating my summary to Dutch. I hope, we will have the opportunity to work together in future too.

Many thanks to Andreas Wombacher for allowing me to use and install the Ubisense system in floor 4. Without his and his group members support, it would have been nearly impossible for me to have some of the data sets that are used in my thesis. I am grateful and privileged for receiving comments on UWB work from Dr. Andy Ward. Thanks to Jeroen from ITBE, for responding immediately to all my silly questions relating to the access points in UT and for other related support like providing floor plans. Thanks to Kees Jan and Paul, for helping me with the Flavour demonstrator and for the help on the PDA-interface of Flavour.

Everyday working atmosphere would not have been so joyful without my colleagues from the PS group. I have enjoyed sharing my office space with Ozlem, Raluca, Mihai, Leon, Georgi, Roland and Hilbrant. It was a highly motivating environment, with a group of extremely smart people around! I thank Georgi for helping me with initial implementation of Flavour. Many thanks to other colleagues and friends- Aysegul (also to Kamil), Yang, Lodewijk, Tim, Ardjan, Stephan, Stefan, Roland, Mohammed, Sinan, Laura, Ricardo, Anka, Michel, Srikanth, Malohat, and Law for giving nice coffee-time breaks and exciting parties!

Ozlem has been my office-mate all throughout these four-and-a-half-years. She has been a fantastic person with such a charming personality. I have learnt a lot of things from her. The time spent with her is full of good memories. She was extremely sweet for having accomodated me in her apartment during the last weeks of my thesis correction. Apart from work, we have always had some wonderful times together partying, dancing, and having lots of fun. Many thanks to Mustafa for his warm friendship and for taking lovely photoshoots at various Indian festivals and parties.

Many thanks to Romanian colleagues cum friends (Raluca/Mihai and Ileana/Stefan) for their support and friendship. Raluca and Mihai are extremely efficient couple and I admire them in many respects. lleana, thanks a lot for being my paranymph and for the wonderful company you have given these years. I hope in future, I will have the time to compete with you in the exotic jewellery making process. Stefan, my past and future colleague, many thanks for all the tips at the time of my writing and also for informing me about the position in Delft. I hope that we will have plenty of opportunity for a fruitful collaboration in future.

My acknowledgement would not be complete without giving special credit to our dancing team of Twente! I admire the excitement shown by Ozlem, Raluca, Nirvana, Ileana and Malohat to learn Bollywood dance and for performing the well-synchronised steps, with Indian costumes at some of the Indian festivals and many other parties at Twente. Thanks so much girls for all those great times and for giving me an opportunity for teaching dance for the very first time!

I also thank Marlous, Nicole and Thelma for their excellent administrative support. I would like to thank the members of the PS, CAES and DIES group for providing me all the support during this endeavor.

I was lucky, no matter where I go, I always find extremely sweet and friendly people around. My stay at Lancaster, in spite of being away from Kiran, was still an enjoyable one. Special thanks to the inmates of “D21” and “D22” – Carl, Yukang, Rose, Flo, Dave, Henoc, Mohammed, Rene, Vasu, Matt, Jamie and Urs – for giving wonderful company during my stay, and for the innumerable party’s, drinks and lunch-sessions. A very special thanks goes to Carl for his willingness to support, be it for the experiments or for discussing certain issues related to work. It is extremely hard to find people who are willing to help to a great extent, without anticipating anything in return. Thanks, perhaps is not enough. Without the help of Carl, I would not have been able to acquire the data that I needed for two of my core chapters. He has responded patiently to all my emails and skype sessions. Carl, I hope that our collaboration would continue in future as well.

Both myself and Kiran are fortunate to have a good shield of Indian community around us, both in Twente and currently now in Eindhoven. Sheela is among the first few Indians I met on campus and I am fortunate to have met a person like her. The instant I saw her, I somehow started to tag along with her. She gave a wonderful support all through these years and especially, while Kiran was battling with his visa-process in India. Sheela, together with Mark and little Thomas, have given us a perfect companionship. I am extremely glad that we still keep in constant touch with each other, and I’m eagerly waiting for you to be back in the Netherlands again.

Vasu-Siva have been such great friends and I am really privileged to have known them. Starting from the support at the very beginning of my arrival in the Netherlands, till my stay in Lancaster, Vasu has been a gift to me. Special thanks to Siva for all the support during my stay in Lancaster and especially for serving hot breakfasts/dinners many times and giving me a lift whenever it rains or gets late! Baby Rohan also gave a great company, pleasure and happiness.

We have been extremely fortunate to have known Supriyo-Anindita. Firstly, I must thank Supriyo for reminding Paul of my application (which was somewhere in a hidden folder), without which I would not have been called for an interview. Over the last few years, we both have shared the unique experience of celebrating our birthdays together! All those birthday parties were simply unforgettable. I have enjoyed having him as my friend and colleague. Anindita, thanks for being my paranymph! I really admire the patience and commitment you have. Whenever, we come to Enschede you both have opened your hearts to accomodate us in your wonderful apartment. I appreciate your kind and constant support. Sweet little Samhita, has been such a perfect little angel bringing lots of joy and fun around, no matter where she goes and who she is with. Special thanks to the strawberry girl, for making Kiran feel extra special!

Many thanks to Pramod-Vishaka (also little Vibhor), Chandu-Meenakshi, Ravi-Madhavi, Sashi-Tomar, Deepa-Vishy, Jay-Sarita, Komal, Ambati, Ragav, Sri, Balaji, Vishnu, Shankar, Salim, Jiten, Sandeep, Srivatsa and many other Indian friends in Twente for their warm support. Special thanks to Vinod-Sowmya for being nice friends, well-wishers and providing warmth and support during our stay in Twente. Thanks a lot for your friendship and I sincerely hope that we all stay in touch, no matter where we end up later in our lives.

The Indian Student Association (ISA-UT) has been a wonderful platform helping students and providing entertainment to both the Indians and International students/staffs of UT. I was very fortunate of being the first board member of that association and for participating in many of the activites organised by the ISA-UT. I thank my fellow board members and friends – Sandeep, Srivatsa, Ambati and Chandu for the commitment and support they have given to this platform. I also thank Shankar, Ragav, Murali, Vishaka for supporting me as co-singer's in many of the Indian events, held here in Twente. It was truely exciting and reminded me of my high-school days.

My deepest thanks to our friends, who are more or less like our extended family in Eindhoven: Sriram-Nandini-little Divya, Rama-Aarthi, Aashish-Purnima, Subbu, Devika-Rajan. They all are the very reason, why we do not want to leave Eindhoven. They have been wonderful well-wishers to us, and gave support to Kiran while I was in Lancaster (and in Enschede). Thanks a lot for your friendship.

I would not have sailed this long journey all alone without the support of my family. The greatest appreciation goes to my father, whose love, help, sacrifice and prayers are something which cannot be measured. Right from the very beginning of my schooling till to-date, he stood by me at all times and totally believed in all my decisions. I still remember all throughout my schooling, he patiently waited outside my school gate with his bike to pick me up every single day for 12 long years! I am extremely fortunate to have him as my dad. My mom, she is no longer with us physically but, she is always showering her love and wishes for us from above. She was a classic example of the finest mom one could ever wish for and would have been extremely happy to see me finish my PhD. We will always miss you Amma, and you will always remain closest in our hearts.

Karthik, my loving brother is a perfect example of an ideal brother! I still cherish all our childhood days. Many days, while I was still awake in the wee hours, he skypes and advise me to get some sleep. Many things, both myself and Kiran, did not have to worry, simply because he took care of all those issues. Together with my sister-in-law Sowmya, they both have been a backbone, giving us all the support and lots of love. Thanks a lot for being there for us.

I am truly gifted to have such sweet grandmothers–both my paternal (patti) and maternal (ammamma) grandma's. Their support, prayers and blessings, especially after we lost our mom was something spectacular and I am deeply indebted to them all my life.

I am extremely fortunate to have found an equally loving, understanding and caring parents-in law for all their wishes and support. Many thanks to my brother-in-law's and co-sister's: Viju-Sudha, Vinu-Divya and Shravan for their support. Vinu-Divya, I'm extremely sorry that I have missed to attend your wedding because of my thesis submission, and I really appreciate your understanding. Many thanks to all my family members (all athai's, chittapa's and Raju mama family) in India for their support and love.

My final, and most heartfelt acknowledgement goes to my husband Kiran. His love, support, encouragement, and companionship has turned my scientific journey into pleasure. The last two years of my PhD, we were separated for longer periods of time– four months away in Lancaster, and the remaining 1.5 years battling between Enschede-Eindhoven every single weekend. Despite all these seperations, Kiran has been the most perfect and understanding husband I could ever wish for. He did a fantastic job designing my thesis cover! It was simply because of him, I had the courage to commence and energy to finish off this PhD.

To end, the last few years have been the best years of my life filled with lots of good memories and lots of learning experiences, which I hope has helped me to evolve a little, both professionally and personally. Deepest thanks once again to all my friends and family for making this journey a pleasant one !

Kavitha Muthukrishnan
August 2009
Eindhoven, The Netherlands.

Contents

List of Figures

List of Tables

Abbreviations

AGPS assisted GPS

AHRS attitude and heading reference system

AOA angle-of-arrival

AR augmented reality

AUV autonomous underwater vehicle

CLM concurrent localisation and mapping

EM electro-magnetic

EKF Extended Kalman filter

EOTD enhanced observed time difference

FDOA frequency-difference-of-arrival

GDOP geometric dilution of precision

GSM global system for mobile communication

GPS global positioning system

GUI graphical user interface

HMD head mounted display

IMU inertial measurement unit

IR infrared

INS inertial navigation system

IEKF iterative extended Kalman filter

ISO International Organisation for Standarisation

IPS indoor positioning system

LOS line of sight

MEMS micro-electro-mechanical systems

MDS multi-dimensional scaling

NIST National Institute of Standards and Technology

NLOS non line of sight

PDA personal digital assistant

PDR pedestrian dead reckoning

RF radio frequency

RFID radio frequency identification

RSSI received signal strength indication

RTLS real-time location systems

RF-TOF RF-time-of-flight

SLAM simultaneous localisation and mapping

SCAAT single-constraint-at-a-time

TOF time-of-flight

TOA time-of-arrival

TDOA time-difference-of-arrival

UAV unmanned aerial vehicle

ubicomp ubiquitous computing

US ultrasound

UWB ultra-wideband

VANET vehicular ad hoc network

VHF very high frequency

VOR VHF omnidirectional ranging

VR virtual reality

WLAN wireless local area network

WiFi wireless fidelity

WSN wireless sensor network

WPI Worcester Polytechnic Institute

ZUPT zero velocity update

CHAPTER I

Introduction

1.1 Localisation and its relevance

Localisation in ancient days referred to navigation – an art of finding the way from one place to another. Tremendous advancement in the science of navigation dates back to the sixteenth century, when instruments like compasses, sextants and the first ever clock to keep the time exactly were devised. Advancement in navigation brought ways and means to explore the world, be it for expansion of the territories or for promoting trade and business. Since then localisation has been explored for several decades as a classical problem in many disciplines:

Navigation systems such as VHF Omnidirectional Ranging (VOR), the ground beacon based air navigation system, have been used since the 1960s by pilots to navigate to their destination. The first satellite based system was the US Navy's TRANSIT system. Operational in 1968, it provided coarse and intermittent two-dimensional positioning for equipment on the ground. TRANSIT's successor, the global positioning system (GPS) [91], improved on TRANSIT by providing more accurate three-dimensional position estimates at a higher frequency. It has been in use since the early 1990s in a myriad of military and civilian applications.

In *robotics*, localisation is typically a prerequisite for exploration, navigation towards a known goal, transportation of material, construction or site preparation. *Autonomy* is the key aspect in *mobile robotics*. In many applications, the mobile robot has an a priori map. Given a map, the robot may localise itself by matching current sensor observations to features in the map. However, usable maps do not always exist, and it is not always possible to have accurate externally referenced position estimates. Most of the robotics research is centered around efficient ways of building these maps, commonly referred to as concurrent localisation and mapping (CLM) or simultaneous localisation and mapping (SLAM) [181, 180] and on solving *data-association* (matching environmental features with features of the partial map) problem.

Precise location and orientation information is a critical requirement in *virtual reality*, which lets the user interact with a virtual environment through the usage of a wide variety of input modalities. The requirements of virtual reality (VR) and augmented reality (AR) systems [31] which places a user wearing a head-mounted display in a partially or completely immersive environment, are applications perhaps requiring the *highest* demands on accuracy and update rates, to prevent users from experiencing motion sickness. The head-mounted displays are typically tracked with accuracies of a *few millimetres in position* and *one degree in orientation*. In addition to catering to high accuracy and update rates, since most of the applications in this category use markers attached to the users body, the system must be *small* and preferably self-contained.

Vehicular ad hoc network [39] focuses on providing ad hoc networking facility to enable vehicle-to-vehicle communication or vehicle-to-roadside-infrastructure communication. In these networks, knowledge of the real-time position of vehicles is a crucial requirement. Research in vehicular ad hoc network (VANET) has resulted in multiple subsets of applications ranging from vehicle collision avoidance to automatic parking. The important requirement for VANET localisation is the *need for infrastructure development* in environments such as in tunnels and urban canyons. VANET is a classic example where a *high mobility* scenario is involved and hence *high update rates and responsiveness* is a mandatory parameter in the design and evaluation of location systems used for VANET applications. This is because slightly outdated positions can be dangerous for certain VANET applications. In addition, for safety applications like vehicle collision avoidance, *high accuracy* is required.

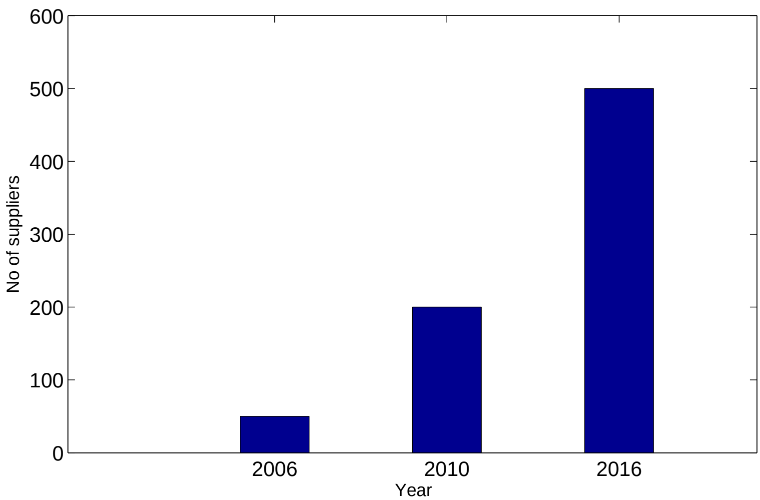
In the early 1990s, Mark Weiser introduced the concept of *ubiquitous computing* [190]–a computing paradigm of the future, where various computing elements will be so seamlessly integrated into the environment that they will be invisible to the user. Several factors have fueled this vision–advances in wireless communication, devices, sensors, hardware technology are paving the way to bridge the gap between the vision and the reality. Closely related to ubiquitous computing is context-aware computing, which provides relevant information and services to the user. Research in the fields of ubiquitous computing and its subset, context-aware computing, has repeatedly highlighted the importance of *location* information as a primary attribute of context. Since then, the research community has responded by developing a myriad of location systems for ubiquitous applications, and many of them have crossed the boundaries of research labs and have gained commercial relevance to-date.

1.1.1 Market growth and trends

The growing need for location systems underscores the importance of addressing this problem –government initiatives to locate emergency calls by cellular network providers and the increasing usage of global positioning system (GPS) in many commercial applications are just a few examples. Since the field is active and vibrant, new services and market players are constantly emerging. For instance, the latest initiative by TomTom [20] reports the location using GPRS to determine traffic conditions and provide real-time feedback to the users. Google have just launched a new service called Latitude [72], which lets smart phone and laptop users share their location with friends and allows those friends to share their locations in return. Latitude uses satellites and cell towers to estimate location. The market growth for GPS products and services alone is expected to grow to US$ 200 billion by 2015 [167].

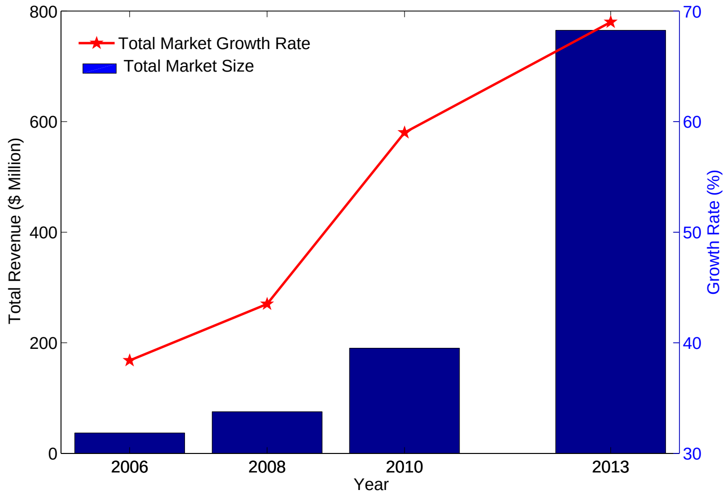
Real-time location systems (RTLS) in various sectors (see Figure 1.1 (a)) drive the penetration of several location-based solutions with different granularity operating both at indoor and outdoor environments. For instance, RFID based systems for locating objects inside buildings dominates the supply chain, especially for Returnable Transport Items [127]–recent announcements by Wal-Mart, the US Department of Defense, Tesco, Marks and Spencer and other large retailers are mandating the use of RFIDs by their suppliers to simplify the supply chain and make savings in efficiency. The market is primarily driven by tracking, locating and monitoring people and things. Also reduction in cost and size accelerates the usage of this technology. Other compelling solutions based on GPS, GSM, WLAN and UWB are also

(a) Penetration of RTLS in various application sectors and some example applications (adapted from: [11])



(b) Trend in number of significant suppliers into parts of RTLS value chain (Source: [11])

Figure 1.1: *RTLS suppliers and market revenue forecasts.*



(c) RTLS market revenue forecasts in millions of US dollars (Source:Frost & Sullivan [22])

Figure 1.1: *RTLS suppliers and market revenue forecasts.*

used in the supply chain and the choice of course depends on the application needs. According to Frost and Sullivan's market analysis [22] the WLAN-based RTLS market is expected to contribute to the largest portion of the RTLS market as enterprises are expected to be able to leverage on the usage of existing WLAN networks. UWB RTLS is expected to be adopted in areas where a high level of resolution is required whereas passive RTLS is expected to be adopted due to significantly lower overall cost. From Figure 1.1 (b) & (c) it is clear that both the global market revenue and trend in the number of RTLS suppliers are increasing exponentially.

1.1.2 Location-aware applications

The development in positioning technologies, with parallel progress of appropriate standards [13] and the falling cost of the technology is increasing the spread of positioning to more and more facets of life. Location technologies have begun to be deployed commercially and are getting integrated with portable mobile devices. In this section we provide a succinct review of some of the key application areas that make use of location information. The stress here is to emphasise how location is at the core of several high-value applications ranging from the time-critical context of emergency response to social networking and gaming.

- **Military:** According to IDTechEx forecasts [11] for RTLS one of the major application domains would be in military (approximately 44% by 2016). Some of the military applications include: battlefield surveillance, mapping opposing terrain, nuclear, biological and chemical attack detection and reconnaissance, sniper localisation, remote border security, target tracking and aid in military training [158, 168].

- **Safety, Security and Fencing:** Camera-based tracking of people is increasingly used both indoors and outdoors for security and surveillance [50]. Security systems may be as simple as low cost, wireless, perimeter security fences, to precise, real-time tracking of intruders throughout a facility [124]. Retailers typically experience an annual loss rate of 15% for shopping carts, which can add up to millions of dollars for larger chains [104]. In a smart kindergarden [175] the progress of children can be monitored by tracking their interaction with toy tops and with each other. Tracking systems based on radio frequency identification (RFID) monitor children at public places like theme parks [14].

- **Search & Rescue:** Positioning and navigation of users (victims) in unprepared environments has been studied extensively. Be it a search and rescue operation to locate lost sailors and downed aeroplanes or allowing rescuers to find people trapped in avalanches [132], localisation is crucial and life-saving. Applications of sensor networks, in particular firefighting has been explored in a range of projects. Tracking of firefighters is proposed for instance based on RFID tags pre-deployed in buildings [135]. Tele-operated robots [204] and a network of distributed mobile sensor systems (on a robot) [111] can be used as well to find victims.

- **Gaming & Entertainment:** Location technologies have begun to be deployed commercially and are getting to be integrated with portable mobile devices. Location-aware gaming takes advantage of these developments and combines social face-to-face aspects of traditional games with the rich complexity that networked computer games can offer. GPS has proved to be a popular platform for wide-area location-aware gaming [61]. HumanPacman [48] uses tangible interfaces and augmented reality. Location-based entertainment pods such as Virtuality 2000SU [195], lets a person seated in a pod wearing a fully immersive head mounted display (HMD) to look around inside a virtual world. Virtual walk-through systems such as the HiBall tracking system are commercially available [194].

- **Social Co-ordination & Smart Networking:** With respect to social co-ordination, Dodgeball [9] lets people use their mobile phone and SMS to advertise where they are and see who else is currently present in different interesting areas. Dodgeball was bought by Google in 2006 [178]. Telecom providers like AT&T [5], offer friend finder applications that let a friend's phone be located. Active campus [1] also makes a map-based buddy list available as a friendfinder application throughout the university campus. Networking is very important, be it for social interactions or collaboration and information exchange. Several systems provide wireless conference devices that are aimed at assisting conference attendees with proximity information. Examples include nTag [15], SpotMe [18] and IntelliBadge [51].

- **Transport & Logistics:** The logistics business is fundamentally all about moving the correct goods from one location to another in the most speedy, reliable and efficient way. Many applications such as tracking high value inventory items or personnel in warehouses, ports or manufacturing plants require a precise location information. Several RTLS solutions – including GPS, global system for mobile communication (GSM), bar code for identification, and other emerging technologies like wireless local area network (WLAN), ultra-wideband (UWB) – are applicable to transport and logistics operations.

The requirements that any location system must meet are tightly coupled to the application needs: *accuracy* ranging from centimetre to metre level, *coverage* ranging from indoor to outdoor, *cost* ranging from high to low. The remaining sections in this chapter outline the primary areas of research in the field of localisation and contributions of our work, and provide an outline for the structure of the thesis.

1.2 Primary areas of research

Localisation has manifested itself as a classical problem in various fields and in this section we highlight in brief, the developments and directions of research in localisation since the emergence of ubiquitous computing. Although the requirements and demands may vary largely among different applications, the core assessment criterion of any location system are *accuracy* – defined as, how much the estimated location deviates from the true location and *coverage* – describes the size of the location systems deployment or working area, while keeping an eye on the cost (infrastructural cost, maintenance and calibration cost). Unfortunately accuracy and coverage of the location systems seldom co-exist and are well correlated with their deployment cost – ranging from easy to deploy coarse-grained systems spanning wider area to expensive, carefully tuned, calibrated, fine-grained systems working in limited deployment area. Hence, even today the research is progressing at an accelerated pace in order to minimise this tradeoff.

Location systems that provide fine-grained information, such as the Bat system [80], are typically operational within the confined area of deployment and hence infrastructure must be deployed large in number, if more coverage is desired. The field gained momentum in 2002 with the Federal Communications Commission (FCC)'s ruling that ultra-wideband devices, particularly suited for high precision ranging, an important phase in localisation, could be operated without a license. This offered alternative fine-grained technology that could work well apart from previously proven ultrasound-based sensing, but covering larger deployment area. Most of the ultrasound [80, 160] and ultra-wideband [182] position systems predominantly rely on accurate timing measurements to estimate position. This fostered the development of several clock synchronisation mechanisms. Location systems detecting the time-of-flight (TOF) of the incoming signal require both the receiving and transmitting entities to be synchronised inorder to determine the distance. This posed a stringent requirement on nodes/sensing devices to include faster clocks capable of nanosecond resolution. Alternative techniques based on *pseudoranging* using time-difference-of-arrival (TDOA) are more attractive and viable, mainly because there is no need for precise synchronisation between the transmitting and receiving entities. All the elements of the receiver can be precisely synchronised using stable clocks at each of the receivers which are periodically corrected via some wired or wireless reference timing signal that is distributed to the receivers.

The uptake of WLANs implemented with radios which make received signal strength indication (RSSI) available to the application software provided an accelerating pace to the field, enabling the location systems to become partially a software rather than an exclusive hardware task. As a result, many research projects and companies have shown interests in providing WLAN-based location systems spanning wider area [32, 12]. But such wide-area location systems do not provide location information of fine-grained granularity without an extensive radio-mapping (calibration) phase. Collecting radio-fingerprints (a process that uses pre-recorded measurements of network characteristics from different locations, to estimate the current location) can be time-consuming, especially if the scale of deployment is large. The radio-fingerprints have to be frequently calibrated to capture any changes in the environment. Usage of automated robots to help collect the fingerprints have been demonstrated for small areas [148], however, neither the effects of robot height and time at which the fingerprint was collected, nor the large-scale practicality of using automated robots to calibrate the fingerprints has not been studied extensively. Currently, many methods are being proposed on how to create training data or radio-fingerprints efficiently.

Fundamental to any location system are the *algorithms* used to estimate location. Position can be estimated based on a vast number of sensing modalities that express range, angle, or some other quantity which can be related to position e.g. infrared light intensity, RFID sightings, WiFi fingerprinting, visible light intensity to infer time of sunset which can be converted to latitude/longitude. Sensor measurements are noisy due to disturbances in the environment and the physical characteristics of the sensor itself. The first stage of many location algorithms is filtering of raw data. There are many different ways to deal with this, ranging from simple averaging to more sophisticated Bayesian filters [65] to reduce the noise and estimate the position. Outlying measurements can also be rejected by making use of statistical methods [189]. Usage of mobility models enables tracking functionality, i.e. providing continuous availability of location information. Lately attention is shifted to algorithm development using heterogeneous and/or multimodal data. This is particularly attractive because no location system provides perfect error-free measurements and can be available at all times. Thus, it is beneficial if measurements from multiple sensing modalities and/or multiple sensing systems can be fused in an effective way.

Hightower [88] proposed a software framework that allows multiple sensing technologies to exist under a single Location Programming Interface. By introducing interfaces between different components the industry has taken off since the horizontal specialisation lets each part of the chain do what it is best at. This has been adopted by research and commercial location systems and has made a significant impact on the field including commercial adoption by Intel [74], research adoption by the PlaceLab project [12], and community adoption through publicly available location estimation library.

Many of the location algorithms require a priori knowledge of the location of anchors or infrastructural beacons. While this assumption seems reasonable for a network covering smaller area, this might be an issue for larger networks (for instance, wireless sensor network (WSN)). Also in some cases, it is possible for the devices which are installed in one place to move (due to mechanical vibrations e.g. slamming doors). While this slight movement may not be significant for certain cases, in some cases small deviations from the original position can cause significant errors in the final location estimate. In such cases, it might be desirable to estimate some of these parameters dynamically while the system is operational. This makes the idea of *autocalibration* attractive. Autocalibration also partially or completely removes the need for people to conduct calibration themselves; calibration for fine-grained systems can be time-consuming and require expert knowledge. The current trend is inheriting some of the existing approaches in other fields such as robotics and tracking in virtual environments [192]. The geometry of the beacon placement is crucial for achieving good accuracy and placement of beacons can be viewed as a cost minimisation problem as they need to maximise the coverage.

Environmental dependence has proven to be a great challenge while designing location algorithms/systems. The nature of the environment influences not only the characteristics of the sensing devices used for localisation but also the magnitude and type of measurement errors. Hence efficient strategies are required both from hardware and from an algorithmic perspective to deal with effects like signal fading and multipath/reflections.

1.3 Thesis Focus

This thesis focuses on formulation of *localisation algorithms* with the capability of fusing readings from multiple modalities. We address the following research question.

How can multimodal localisation be achieved and what performance improvements can it offer?

Here, "multimodal localisation" refers to some combination of observations gathered using different (heterogeneous) sensing modalities. As mentioned in Section 1.2 one of the core assessment criterions of any location system is accuracy. While our main focus is on improving accuracy, we also highlight the benefits multimodal localisation can have on improving other desired properties such as coverage, less density of infrastructure support, and improved update rates.*

We hypothesise that there are plenty of ways to improve location accuracy by combining different modalities and, regardless of the type of data, incorporating multiple modalities would improve the accuracy of the resulting location estimates. The methods can range from simple smoothing and filtering to fusion and tracking. *Fusion* typically refers to the effective use of two or more heterogeneous sensor observations to determine location and *tracking* offers the capability to provide continuous stream of location estimates, even amidst the absence of input observations. While smoothing of location estimates can be one easy way to improve the quality of the final estimate, fusion and tracking are sophisticated ways to improve the accuracy. We illustrate by collecting different types of measurements – ranging from simple and easily available RSSI to complex timing information (such as TOF or TDOA) or angle information (angle-of-arrival (AOA)) from a wide variety of popular technologies today (WLAN, ultrasound, UWB and inertial sensors) and in every instance we highlight the benefits of smoothing, filtering, fusion and tracking. Another important aspect of the work

*However, not all these effects have been thoroughly evaluated in this thesis.

presented in this thesis is the evaluation of algorithms by gathering data from real deployments. Experimentation is a valuable tool for testing the performance of any algorithm as it prevents the unrealistic assumptions and validates application with real sensors. Ultimately location systems are to be deployed in the real-world, hence data used in algorithms must capture the physical effects like multipath, obstruction etc. that are present. This work is more oriented towards the formulation of algorithms that are capable of fusing multimodal data and in comprehending the benefits by applying the concept of multimodal localisation on a large variety of measurement types rather than performing specific optimisations (such as, complexity or sensitivity analysis) to the algorithms themselves.

With regard to the previously mentioned areas of research and focus of this thesis, we enumerate in the following the main contributions of this thesis.

1.3.1 Contributions of this thesis

1. **Taxonomy and survey of location systems**

   We begin by systematically analysing the basic principles of localisation through a review and classification of the state of the art. From our detailed survey, it is evident that no location system is error-free and suited for all situations. For example, pure inertial sensors suffer from drift, ultrasound sensors require clear line of sight and magnetic sensors are affected by ferromagnetic and conductive materials in the environment. Thus, we rationalise *multimodal localisation* as one of the promising ways for improving location accuracy and robustness. Apart from improving performance of the location system in limited measurement volumes, fusion of heterogeneous sensing systems will ultimately allow people to move from place to place without loss of location knowledge.

2. **Characterisation of raw measurements**

   We investigate what improvements in accuracy can be achieved by fusing multimodal data. The particular quantitative improvement in estimation that results from using multiple sensors depends on the performance of the specific sensors involved (data collection rates, observational accuracy), environmental effects, and the specific algorithms used for data fusion. Data characterisation allows us to comprehend the benefits of fusion. Additionally, characterising strength and weakness of the data, will enable appropriate choices in selecting different modalities for improving the accuracy. One of the other merits of characterisation is that some positioning algorithms require known error distributions to function effectively. All measurement characterisation is performed on available sensors/technology that are used commonly (as a research prototype or commercial product) for localisation.

3. **Algorithms for inferring motion and location from WLAN RSSI**

   We present novel algorithms to infer movement that make use of inherent fluctuations in the signal strength. The goal is to demonstrate how simple location algorithms like

Centroid or Weighted centroid could benefit from knowing the motion of the device to be located and by using history of past location readings to improve accuracy. The solution we provide could be viewed more like smoothing of location estimates based on motion derived from RSSI and as a result eliminates the so-called "teleportation effect" that commonly occurs in location algorithms using RSSI data. To the best of our knowledge, motion models are normally used only in probabilistic algorithms and simple deterministic algorithms have not used a motion model in a principled manner. We evaluate the performance of the algorithms against traces of RSSI data collected from different environments.

4. **Positioning algorithms using heterogeneous data**

   While smoothing of location estimates can be one easy way to improve the quality of the final estimate, fusion and tracking are sophisticated ways to improve the accuracy. We demonstrate the benefits of fusion and tracking on sophisticated data such as the time-of-flight, angle-of-arrival and time-differences-of-arrival measurements.

   - We address the benefit of fusing heterogeneous observations (pseudoranges and angles) gathered from an ultra-wideband system. We present positioning algorithms that are based on error minimisation approach and state-estimation approach using heterogeneous data collected from two different deployments – mimicking the low-overhead deployment vs. carefully planned and calibrated deployment. We demonstrate that the presented algorithms can work with perfect and imperfect data and highlight the impact of calibration on accuracy of the location estimates. We also consider the implications of using just one type of data to show the significant merit of adding heterogeneous observations.
   - We present navigation and tracking solutions using a combination of ultrasound and pedestrian dead reckoning methods. The position error in a purely inertial system increases with time and requires correction from external sources. We address this problem by deploying ultrasound sensors as landmarks correcting for the inertial drift. We present algorithms to support tracking and navigational guidance. A thorough evaluation using measurements gathered from real deployments is performed.

1.4 Thesis Overview

In the next chapter we present our taxonomy and describe the basic principles in localisation. We then explore the current trends in commercial products and research in the area of localisation and provide motivation for the topic addressed in this thesis. This chapter corresponds to **Contribution 1** and is an expanded version of a paper published as [8][†]

Research in localisation is tightly coupled to the requirements from applications. In Chapter 3 we set the scene by choosing three specific applications covering both infrastructure-based positioning and ad hoc-based positioning systems that are of direct relevance to the

[†]For author's publications, refer to page 173 of this thesis.

work presented in this thesis. We show how our presented taxonomy can be applied to identifying the requirements of the three chosen applications.

We highlight the importance of characterisation of raw measurements through our analysis performed in Chapter 4 – 6 which corresponds to **Contribution 2**.

Chapter 4 addresses algorithms to detect movement by leveraging existing WLAN infrastructure. We illustrate the benefit of smoothing and give detailed methodologies on how we achieve better accuracy by incorporating a motion model into common deterministic algorithms. This corresponds to **Contribution 3**. This chapter subsumes five publications [1,4,5,6,7].

Chapter 5 and Chapter 6 focus on positioning algorithms that are capable of fusing heterogeneous data which corresponds to **Contribution 4**. Specifically, Chapter 5 deals with the formulation of positioning algorithms to work on heterogeneous data (pseudoranges and angles) gathered from an ultrawideband system. This chapter is joint work with Lancaster University and has been published as [2].

Chapter 6 addresses formulation of navigation and tracking algorithms using a combination of inertial sensors and ultrasound sensors. A part of this chapter has been published as [3] and is a result of joint work with Lancaster University.

Finally, Chapter 7 presents the conclusions that can be drawn from this research and briefly discusses possible directions for future research.

CHAPTER II *

A Taxonomy and Survey on Location Systems

Localisation is a classical problem and has been studied widely in many different domains resulting in a bewildering number of location systems exhibiting different characteristics. This chapter defines a taxonomy by examining issues involved in the design and evaluation of location systems. We then survey a variety of location systems and discuss the basic parameters such as technologies and methods used to estimate location, achievable granularity, the update rates they provide, their cost in terms of the device being localised as well as the infrastructure required, and their privacy implications. The objective is to understand the underlying principles of a variety of approaches used to gain location-awareness, to comprehend the many tradeoffs involved in designing location systems and to explore the current trends in commercial products and prominent research in this area. To end, we rationalise multimodal localisation as one of the promising ways to go for improving the performance.

*This chapter is an expansion of the paper published with the title, *Towards Smart Surroundings: Enabling Techniques and Technologies for Localization*, In the Proceedings of the International Workshop on Location and Context-Awareness (LoCA), co-located with Pervasive, Munich, Germany, May 2005 [138].

2.1 Definition and Preliminaries

In this section we overview the basic definition and concepts. The conventions used here are important as they form the basis for what comes later in this thesis.

Localisation and Tracking:

Localisation (synonymously location estimation, location sensing, positioning)* and *tracking* are terms that are used widely in literature. Although they represent similar concepts, there are subtle differences between the two. *Location estimation* is referred to techniques enabling a mobile or static object/node/device to answer the question "Where am I?". This means that the *object* needs to find out its location relative to the environment in which it is present. It might be relative to a map or to another node, or to a global coordinate system. Tracking is often coupled to location system where the computation is done centrally. The tracking system can monitor or track objects that are being localised there by providing a continuous stream of location estimates. In contrast, positioning enables an object to be located at an instance of time. Both tracking and positioning systems are implicated for issues such as scalability and privacy. An example of a tracking system is the Active Badge [186] and Active Bat systems [80]. GPS positioning comes under the category of positioning systems.

Location System Entities:

The entities enabling any location sensing system can be broadly classified as *Infrastructure entities* and *Mobile entities*.

- *Infrastructure entities* consist of components that are present in the environment assisting the location estimation. For instance, in case of a GPS positioning system, the satellites form the infrastructure and in case of a WLAN positioning system, the wireless access points form the infrastructure. An assumption in most of the location system is the availability of infrastructural entities called as the *anchor nodes* or synonymously called *landmarks* or *beacons* [170].

- *Mobile entities* are those whose location needs to be determined. In most cases, mobile entities are either tag or marker-based. GPS receivers are examples of mobile entities. It is possible that the moving device acts either like a transmitting device initiating the location estimation process or as receivers when the infrastructure components act like a transmitting source. However this is purely dependent on the type of architecture and requirement from the application.

The presented two categories can also vary among different location systems. While some location systems like GPS [152] and Cricket [160] use both the infrastructure and mobile components, certain other location systems like some of the camera tracking systems perform the estimation in an unobtrusive manner, meaning that the object wishing to be located, does not need to carry a tag or any markers. This might on one hand be advantageous, while on the other hand it also intrudes the privacy.

*Positioning can actually refer to the actual position (in terms of coordinates) while location often represents symbolic location like room, building. We, however, do not make any distinction here and use them interchangeably throughout this thesis.

In some cases, there are location systems that use only the mobile entity, without the usage of infrastructural support. Those systems can be grouped under the category of *autonomous positioning systems*. Inertial sensing systems are an example. Such systems are highly useful in disaster recovery applications in particular as assuming the presence of an infrastructure in the advent of fire or an earthquake is not a realistic assumption. In mobile robotics and virtual reality applications autonomous positioning systems which are capable of locating itself and the creating the map of environment are popularly called *SLAM*, Simultaneous Localisation and Mapping systems.

2.2 Taxonomy

Researchers have classified location systems based on a number of aspects. Unlike the earlier presented taxonomies that are based on evaluation properties [89, 138] or based on functional methods [101], the taxonomy we present here includes both methods and observed spatial phenomena used to compute the location and the evaluation properties used to assess the performance and design of any location system. We build our taxonomy based on the following eleven criteria listed with definitions in Table 2.1. In this section we explain in detail our eleven defined classifications below (one may note that it is possible to sub-categorise them further).

Criterion	**Definition**
1. Dependency	Support/no support from the infrastructure
2. Coverage	Size of the deployment area
3. Range	Range of beacons
4. Architecture	Whether locatable acts as a transmitter or receiver
5. Computation	Where the location computation is performed
6. Output representation	Type of location provided
7. Object association	Use of marker/tags for localisation or not
8. Measurement type	Type of gathered observations
9. Technology used	Type of the modality used
10. Estimation method	Method for predicting locations from the observations
11. Performance measure	Properties used to evaluate the location system

Table 2.1: *Criteria used for the taxonomy and their definitions.*

1. Dependency

Any location system can be broadly categorised into either *infrastructure-based* or *ad hoc-based* positioning methods depending on the type of entities that form the system (as explained before). Infrastructure-based systems are those that require support from the devices present in the infrastrucure to aid the location estimation process. Infrastructure-based location systems use anchors/beacons placed at known points in the environment to determine the positions of other objects in that environment. On the other hand, ad hoc-based methods facilitate autonomous positioning capability. The typical use case of infrastructure-based positioning systems include: tracking of pallets within warehouses, finding colleagues in a work environment, for logistics and transportation, etc. while ad hoc positioning systems are more desirable for applications such as disaster recovery, emergency response, etc.

2. Coverage

Coverage describes the size of a location system's deployment or working area. Coverage can be broadly classified as *indoor* and *outdoor*, or can be sub-classified as *building, campus, city,* etc. Some of the systems are limited to this coverage because they assume the installation of a special infrastructure or must have knowledge about the physical layout of the building and beacon information. Scale is also another term to represent coverage.

3. Range

Closely related to coverage is the range. Any location system can have a good coverage only if they are within the range of a beacon. We categorise them as *short-ranged*, *medium-ranged* and *long-ranged*. By short-range, we mean anything less than 10 m, medium-range varies between 10-100 m and anything above is long-ranged. Systems with short-range beacons have difficulty in covering wide-area.

4. Architecture

It is possible to have two different location architectures – *active* and *passive*. In an active architecture, the transmitter is attached on the object that needs to be located and periodically broadcasts messages. Receivers are deployed in the infrastructure to listen for such broadcasts and estimate the location of the transmitting device. In contrast, in the passive architecture, the role of transmitting and receiving devices is reversed. The beacons are deployed at known locations and periodically transmit their location to the mobile device to estimate its distance to every beacon that it hears and to estimate its location. The infrastructural beacons that are deployed for other purposes (e.g. communication) that are now being utilised for positioning purposes can fall under either active or passive architecture depending on the beacon's role of acting as a transmitter or a receiver. Examples include GSM base stations, WLAN access points etc. The architecture of a location system influences its *scalability*, and *user privacy*, *ease of deployment* and *device-tracking performance*. Active architectures are particularly weak in terms of preserving user privacy as the infrastructure can track the users.

5. Computation

It is also possible to categorise location systems based on where the actual computation is performed as *centralised*, *distributed* or *localised*. Synonymously, *network/node* localisation is described in some cases in sensor networking [105] and cellular-based location systems. The method of computation influences the location system's *scalability*, and *user privacy*, *ease of deployment* and *device-tracking performance*.

6. Output representation

Output representation typically denotes the type of location information provided. As Hightower [89] defines, location representation can be either *absolute* or *relative* or can provide *physical* or *symbolic* location information. Absolute location systems are those which use the same coordinate system for every device. GPS is a classic example under this category which describes the position of all the GPS receivers in terms of latitude, longitude and altitude. In contrast, relative location systems are characterised by different coordinate systems for different mobile devices. Two devices placed at the same location may provide different readings depending on which frame of reference is used. Translating between relative and absolute location estimates can be performed if knowledge about the relationship between both coordinate systems is available. Physical and symbolic systems classify location as a description of place. Physical location (e.g. latitude/longitude) is equivalent to the concept of position while symbolic refers to location description in human understandable form (e.g. near the coffee room).

7. Object association

As defined by Ward [189], location systems can be classified based on method of sensing as *tagged* and *untagged* systems. Tagged or marker-based systems (either passive or active) are associated with an object, thus allowing objects position to be determined, while untagged systems allow to estimate the position of the objects directly. Tags are active or passive objects which are designed to emit or display a known pattern which can be detected by the location system's sensors (e.g. includes ultrasound or UWB transmissions and barcodes). Although tagging objects does increase setup complexity, as all objects that are to be tracked need to be augmented, it does have some benefits. By augmenting the objects, two objects that might otherwise look identical can be individually identified. Also, it generally decreases the amount of computational power required for recognition and tracking. Untagged systems are those which do not require augmentation of locatables and are thus unobtrusive. An example of this is a camera-based tracking system that employs facial recognition.

8. Measurement type

All location systems are based on some models from the physical world. The process of finding a location needs to utilise relationships between the position and physical properties of the space. There are different types of measurements such as angles, distance, time, signal strength, connectivity, etc. We outline some of the most common observation types utilised in location systems in Section 2.3. Measurements[†] can be grouped under two categories – *range-based* and *range-free*. Range-based schemes make use of certain specialised hardware to infer the spatial relationship between the object that needs to be located and the infrastructural components. On the other hand, range-free schemes do not use specialised hardware, but make use of other metrics like connectivity and signal attenuation to infer distance.

9. Technology used

Location systems need to have some kind of physical media to establish spatial relationship and compute location or position from the observed phenomena. Most widely used type of technology are infrared, ultrasound, radio, optical, electromagnetic. Section 2.4 describes them in more detail, listing each of technology's merits and demerits.

[†]Throughout this thesis, measurements, data and observations are used interchangeably.

10. Estimation methods (Localisation algorithms)

A location estimation process needs to combine the technology and the observation (or measurement) model to estimate the location. For a majority of the measurement types ‡ typically one of the following three methods are used to estimate location. The first approach uses a simple geometric model to calculate intersection of circles (range), lines (angles) or hyperbola (pseudoranges). These simple algorithms do not account for measurement error and cannot make use of optimal redundant data which overspecifies the solution. The second approach is based on *optimisation algorithms* that are specifically designed to find a solution minimising the total error between the collected data and the location estimate. However, they do not make use of past estimates in predicting future estimates. *State-estimation* algorithms iteratively combine the previous estimate of the state with the observed measurement type. We detail the prominent estimation methods used in Section 2.5.

11. Performance measure

A location system can be evaluated in numerous ways. It can primarily be evaluated based on the accuracy of its estimates, but other properties such as high update rate, support for locating mutiple objects, latency, privacy awareness, cost (computation, maintenance, setup, calibration, etc.) are also of importance to many applications. We list the various performance parameters in detail in Section 2.6.

Note, that it is possible to have a mix of features from the presented taxonomy. For instance the system with the centralised architecture is the one which facilitates tracking easily.

‡TOA,TDOA, RSSI or AoA

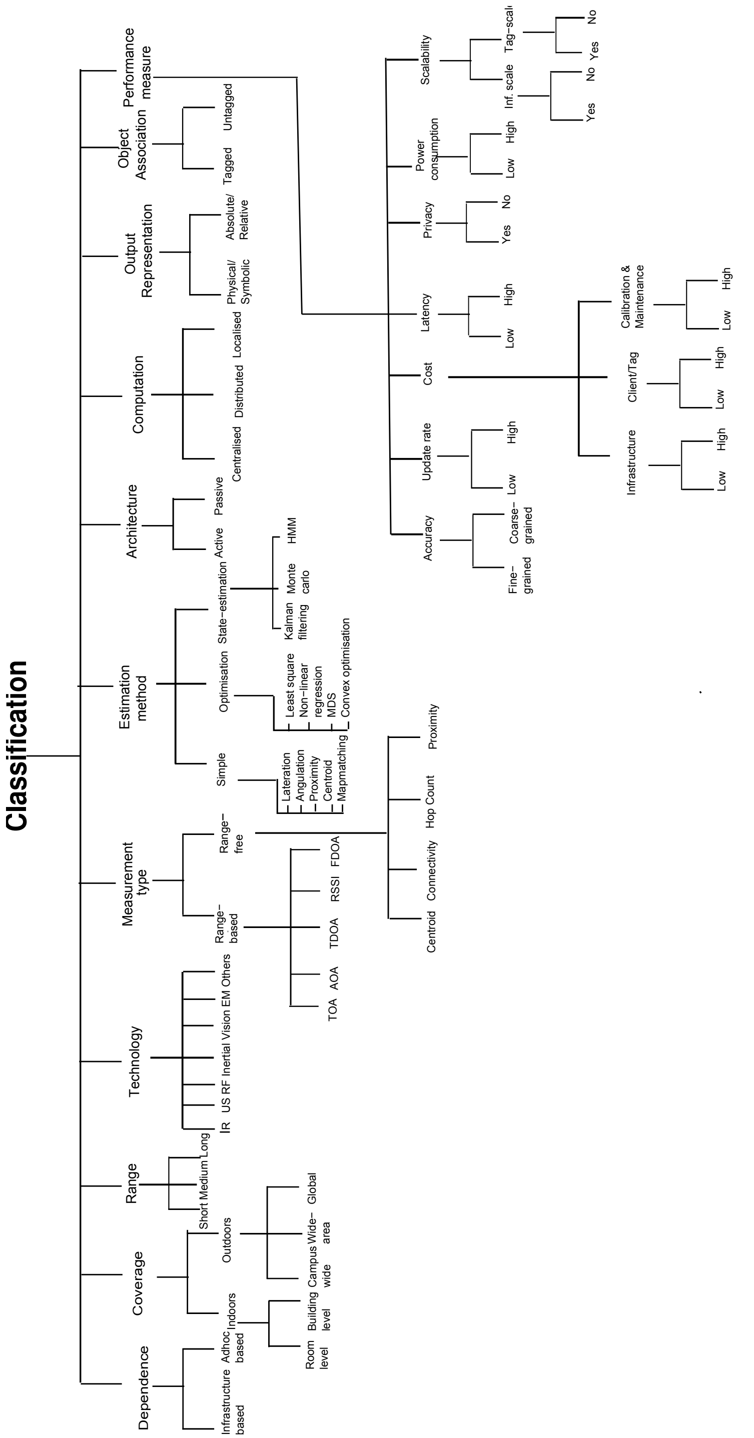


Figure 2.1: *Taxonomy of location systems, based on eleven classification criterions defined in Table 2.1.*

2.3 Measurement/Observation Type

Range-based

The *ranging technology* forms the heart of any range-based location system. There are several range-based techniques such as time-of-flight (TOF), time-differences-of-arrival (TDOA), angle-of-arrival (AOA), and received signal strength (RSSI). The TOF and TDOA make use of signal propagation time for finding the range or distance. AOA uses angular estimates instead of range/distance estimates. Angles can be estimated using rotational directional beacons or by phased array antennas [169]. RSSI makes either theoretical or empirical calculations to convert the signal strength measurements to distance estimates or can employ fingerprinting-based solutions.

- **TOF:** The TOF measurements are employed on signals that propagate at the speed of light or speed of sound. In many location systems, the receivers detect the TOF of the incoming signal. If the receiver knows the precise time of transmission (t_t) of the signal and the speed of the signal through the environment (v), it is possible to determine the distance between the receiving device and the transmitting device using the following relation:

$$d = v(t_a - t_t) \tag{2.1}$$

  In practice this technique can only be used if the nodes/sensing devices are equipped with fast clocks capable of nanosecond accuracy. In addition there should be synchronisation between the transmitting and receiving entities. Typically the receivers do not know the precise time of transmission of the signal from the transmitting device. Achieving synchronisation of the transmitting and receiving entitites requires expensive and power-hungry circuitries at the transmitting device. If synchronisation is not feasible with high accuracy, estimate of the range can be obtained by measuring the round-trip time of flight. However, this also requires a precise evaluation of the time used by the target node to process the message and send a reply back to the transmitter. The use of faster clocks increase the cost of the hardware and power consumption as well.

- **TDOA:** *Pseudoranging* using TDOA is more attractive mainly because there is no need for precise synchronisation between the transmitting and receiving entities. All the elements of the receiver can be precisely synchronised using stable clocks at each of the receiver which are periodically corrected via some wired or wireless reference timing signal that is distributed to the receivers. This means that, since receivers are synchronised an event that is determined by one receiver as occuring at time t, it would then be considered by other receivers as occuring at the same time. It is then possible to obtain the information about the position of the transmitting device by comparing the differences of the signals time-of-arrival at multiple receivers. Supposing a signal was transmitted at an unknown time t_t and received at times t_{a_1} and t_{a_2} at receivers 1 and 2 respectively. Then the difference between the distances is calculated using the following relation.

$$d_{\Delta_{12}} = v(t_{a1} - t_t) - v(t_{a2} - t_t) = v(t_{a1} - t_{a2}) \tag{2.2}$$

Note that the unknown transmission time is not required to determine the distance difference.

- **AOA:** AOA-based techniques measure the direction from where a signal was emitted. AOA requires antenna arrays lined up in accurately known patterns. In order to calculate a location in three dimensions, both the azimuth angle and the elevation angle should be measured. Note that there is need for synchronisation between receiving entities in the AOA approach. By using a 2D antenna array, a single receiver unit can determine the bearings of the transmitted signal in both azimuth and elevation.

- **RSSI:** The RSSI-based technique measures the signal power received by a wireless device. radio frequency (RF) propagation in the free space follows the Fris equation [81] and the path loss is proportional to $1/d^2$, where d is the distance between the transmitter and the receiver. In reality, the path loss is harder to predict since it depends on the characteristics of the environment where the communication takes place. Despite the unpredictability of the radio signal propagation, several researchers have used RSSI to calculate location. The biggest advantage of RSSI over *time-based* or *angle-based* methods is that they can be implemented on existing wireless systems with little or no hardware changes. The modulation method, data rate and system timing precision is not relevant. Coordination and synchronisation between the transmitter and the receiver is not required. But on the negative side, RSSI methods are highly affected by interference and multipath on the radio channel. Therefore, the accuracy achieved is usually much lower compared to time-based techniques. There are two ways in which RSSI can be used to estimate location: radio propagation modeling and radio fingerprinting. Kjaergaard [101] presents a detailed review on fingerprinting methods. The effectiveness of such solutions in real-world applications remains difficult to evaluate since RF propagation across different environment varies greatly.

- **Doppler shift:** Doppler shift is a mechanism that is observed when the relational velocities of a number of beacons change with respect to a mobile device. The beacons can be mobile or stationary [125] but their location and velocity must be known. This is also known as frequency-difference-of-arrival (FDOA) and is analogous to TDOA for estimating the location of a radio transmitters based on observations from several known points.

Range-free

Range-free algorithms overcome cost and system complexity by making use of solutions that do not rely on dedicated hardware for measuring distance or angle. The location of each node is estimated by exploiting proximity information. Range-free techniques employ algorithms that calculate the distance in terms of hop count to beacon nodes [85]. The advantage of these schemes lies in their simplicity, as sensors do not need to use TDOA, time-of-arrival (TOA), AOA or RSSI measurements and simpler schemes can be employed. Some algorithms that use range-free schemes are Centroid algorithm, DV-Hop, DV-Radial, DV-bearing, Amorphous, Point-In-Test (PIT) and Approximate Point-In-Test (APIT) [85]. Most of these algorithms are specifically designed to enable localisation in wireless sensor networks (WSN).

2.4 Technologies

The physical medium forms the heart of any location system. Depending on the required range, propagation speed, cost, precision, bandwidth, etc. one can choose the required technology for a specific application. Most commonly used technologies are infrared-based [186, 107], ultrasonic-based [160, 26, 82], acoustic-based [70], electromagnetic-based [16], inertial-based [199], optical-based [52], and radio frequency-based systems [157, 32]. Other technologies like using powerlines available at home are presented in [153]. Depending on the type of frequency range used, radio frequency can be categorised into for example, RFID [145, 126], WLAN (IEEE 802.11b) [32, 202], Bluetooth (IEEE 802.15) [130], FM transmission towers [108], wireless telephony infrastructure such as GSM [151] and ultra-wideband [182] (UWB is based on sending ultra short pulses typically <10 ns). In this section, we present the most prominent technologies by outlining the principle, requirements, advantages and disadvantages.

Infrared-based Infrared (IR) signals are light waves outside of the spectrum that is visible to humans. Infrared signals need a direct line-of-sight, since they are blocked by walls and most other opaque materials. Infrared signal travels at the speed of light and are useful in applications where there is no requirement for fine-grained location data. One common problem with infrared systems is that they will fail in case the tag is covered up by a shirt or jacket. These systems can provide locations with room- or metre-scale granularity.

Ultrasound-based Ultrasound (US) does not operate in the radio spectrum, but uses sound waves instead. Since ultrasound signals travel at the speed of sound (around 343 m/s), relatively much slower than the RF, making precise TOF or TDOA measurements (microsecond resolution) for ultrasound (US) signals is much easier than for RF or infrared signals which travel at the nearly speed of light. This rate of travel makes it possible for inexpensive electronic timers to measure propagation delays of ultrasonic signals. Although the exact speed of ultrasound is affected by environmental conditions, the variations can be modelled and the resulting speed of sound can be predicted accurately. If the time at which the ultrasonic signal leaves a transmitter is known, then the transmitter-to-receiver distance can be calculated directly by measuring the time-of-arrival of the signal at the receiver. An intrinsic property of ultrasound is that it does not penetrate solid objects, such as walls, tables and doors. On the one hand, this can severely limit the range of transmitted signals, especially in small areas, but this also means that a received signal was probably transmitted from the same room. Therefore, ultrasound offers a reliable way to determine the current space an object is in. A feature of ultrasound is that since it takes time to travel from a transmitting to receiving unit, it results in a *lag* (typically in the order of *ms*) in location readings. Though this lag is not really a problem for most of the applications, it limits the usage of ultrasound for augmented/virtual reality applications on the grounds that the lag can cause motion sickness [94]. Ultrasonic location systems have the capability to provide accuracies of several centimetres and aggregate update rates in the tens of Hertz. The mobile tags must have line-of-sight to the fixed receivers, similar to other tagged systems, with the exception of radio-based systems. Broadband US systems have the capability to track multiple objects simultaneously, reduce noise and offer higher update rates [84].

Narrowband Radio Frequency (WLAN, Bluetooth, Zigbee, GSM)-based Radio frequency is very attractive mainly because the signals can pass through solid objects. Depending on the frequency used, they can be further classified as either *narrowband* or *wideband* RF systems. WLAN, Bluetooth, RFID fall in the narrowband category, while UWB systems come under the other category. The main advantage of a location system using GSM or WLAN signals, is that they can leverage the personal devices existing hardware, there by limiting the cost of the tag (client's device) and the infrastructural cost. Another inherent advantage is the ability to pass through solid objects. In cluttered indoor environments, the signal reflections can impose severe problems. Since the radio waves travel at the speed of light, highly precise clocks are required to make timing of the signals precise, thus making the RF systems utilising time quite restrictive (due to hardware cost and power consumption). Most of the RF-based systems predominantly use RSSI-based methods. Location systems based on conventional narrowband RF technology work coarsely indoors because they are plagued by multipath distortion caused by radio signals reflected from walls, desks, people and equipment. This can often lead to positioning errors of several metres.

Ultra-wideband-based The term ultra-wideband (UWB) in general refers to any radio technology with a bandwidth larger than 500 MHz. However, UWB is also the name of a standard that has recently (March 2007) been approved as an International Organisation for Standarisation (ISO) standard [25], and refers to a high-speed data transmission protocol operating in the frequency band between 3.1 and 10.6 GHz.

UWB radio positioning systems can be accurate to about 6 inch (15 cm) indoors because they are much less affected by multipath distortion than conventional narrowband RF systems (because accurate timing in the order of nanoseconds yield accurate pseudoranges and better reflection rejection) and the position estimation is based on time-of-arrival rather than signal strength. The advantages of UWB are its ability to pass through objects such as walls/clothing and due to the higher frequencies, it can cope with the effects caused by multipath better than other RF technologies. The higher frequencies lead to shorter radio pulses, making it easier to determine which signals are correct and which are the result of multipath since the original pulse and its reflections are less likely to overlap each other at the receiver. However, if the direct path signal is blocked, the first arriving pulse is a reflection.

Electromagnetic-based Electromagnetic (EM) tracking devices function by measuring the strength of the magnetic fields generated by the sending current through three small wire coils, oriented perpendicular to one another [16]. These three coils are embedded in a small unit that is attached to the object the system needs to track. The current has the effect of making each wire an electromagnet when flowing through it. By sequentially activating each of the wires, and measuring the magnetic fields generated on each of three other perpendicular wire coils, it is possible to determine the position and orientation of the sending unit. Electromagnetic systems can be accurate to a few *millimetres in 3D* however, they are affected by large metal objects, and require calibration to achieve good accuracy in a building containing steel enforcements. They rely on an infrastructure of wire coils that must cover the area of tracking and are therefore difficult to install in rooms with high ceilings and even more difficult outdoor. The advantage is the possibility of providing hundreds of updates per second even while tracking a number of subjects. The disadvantages include an limiting operating range of 2-5 m and best results are achieved if the operating volume is an open area. Calibration must be performed if tracking is to be accomplished in typical indoor environments, otherwise performance can be impaired by occlusions caused by furnishings, which typically have conductive or magnetic components. These tracking units may experience interference operating in the vicinity of other devices that produce magnetic fields, as well as metal objects, such as office furniture, that disrupt magnetic fields. The systems based on electro-magnetic (EM) are often prohibitively expensive.

Inertial-based Dead reckoning –a process of estimating one's current position based upon a previously determined position– has the distinct advantage of providing autonomous positioning capabilities. Inertial-based systems measure the changes in the motion of an object being located thereby, requiring no support from the environment. Typically they use 3 axis-accelerometres and 3 axis-gyroscopes to determine the change in velocity and direction. Integrating readings from gyroscopes once, leads to angular velocities and integrating the acceleration twice will result in the position to be computed. They require initial position and orientation. However, positions provided by this method will unavoidably drift over time due to errors in measurements being integrated [66]. Hence the position error in a purely inertial system increases with time and requires correction from external sources. A common practice is to periodically use external sources to correct position estimates [165]. Typically inertial sensors offer high update rate in the order of hundreds of hertz.

Contact-based Contact-based systems are physically limited in scope, because they require the entity being tracked to be in direct contact with the sensing device. Scalability is a major concern for systems based on load-sensing and pressure-sensing [150]. Typically, a matrix of load or pressure sensing cells are deployed underneath the floor tiles, and by analysing the distribution of weight across the deployment area the object can be located.

Vision-based Location systems that process images from cameras in order to locate people and objects are classified as computer vision-based systems. The systems, however, still require significant computational power and resource for image processing and feature extraction. Note that the algorithms used are not conventional localisation algorithms, but more related to image processing and feature extraction. Many of the vision-based system do not require tagging the object to be located [50]. But there exist vision-based systems with markers [35]. The major technical limitations are sensitivity to changes in lighting and ability to track multiple objects accurately. In addition, privacy still remains a major concern and one of the major constraints for wide-scale adoption [35, 42].

Table 2.2: *Enabling technologies.*

Technology	Merits	Range	Remarks
Infrared	Components are cheap and ubiquitous Compact Low power	Typical range is upto 5m	Restriction to line of sight conditions Unusable in direct sunlight
Ultrasound	Relatively slow propagation (speed of sound) Allows for precise measurement at low clock rates, making the system simple and inexpensive	Typical range is 3 – 10m	Environmental factors have substantial effects
Radio Frequency	Better than IR in terms of bandwidth, cost and speed	Typical range: Bluetooth: 10–15m WLAN: 30–100m RFID (passive): 1–2m RFID (active): 10–100 m GSM: > Tens of km	No proper propagation model exists Affected by multipath
DC Electromagnetic	High precision High signal propagation speed	Typical range is 1–3m	Signals are sensitive to environment Precision calibration required, hence expensive Difficult to install
Optical	High precision Compact Low power No tag required (in some cases)	Limited Range (few metres)	Restriction to line of sight conditions Unusable in direct sunlight
Inertial	Ad hoc positioning capabilities	n/a	Errors accumulate over time Calibration
UWB	High precision and accuracy Less affected by multipath than the traditional RF systems Low power	Range 100 m	Expensive Higher receiver density than the conventional RF systems

2.5 Location estimation algorithms

In order to calculate a location based on distance estimates (or angles/signal attenuation) using previously explained *technologies and measurement types*, an algorithm for location estimation is required. We categorise most of the location estimation methods to fall under one of the following three categories–(i) algorithms without optimisation (simple algorithms), (ii) optimisation or error minimisation algorithms and (iii) state estimation algorithms. By simple algorithms, we refer to algorithms that use simple geometric properties and that do not take measurement errors into account, and cannot make optimal use of redundant data which overspecifies the solution. By contrast, the second approach is to use optimisation algorithms which are specifically designed to find a solution minimising the total error between the collected data and the location estimate (i.e. the residual error). Basically, they iterate the solution space and compute expected measurements for each estimate of the solution. The algorithms that utilise the solution state (either current state, or current and past states) can be grouped under the third approach of *state-estimation algorithms*.

Algorithms without optimisation (Simple algorithms): Figure 2.2 illustrates the working principle of some of the algorithms that come under this category.

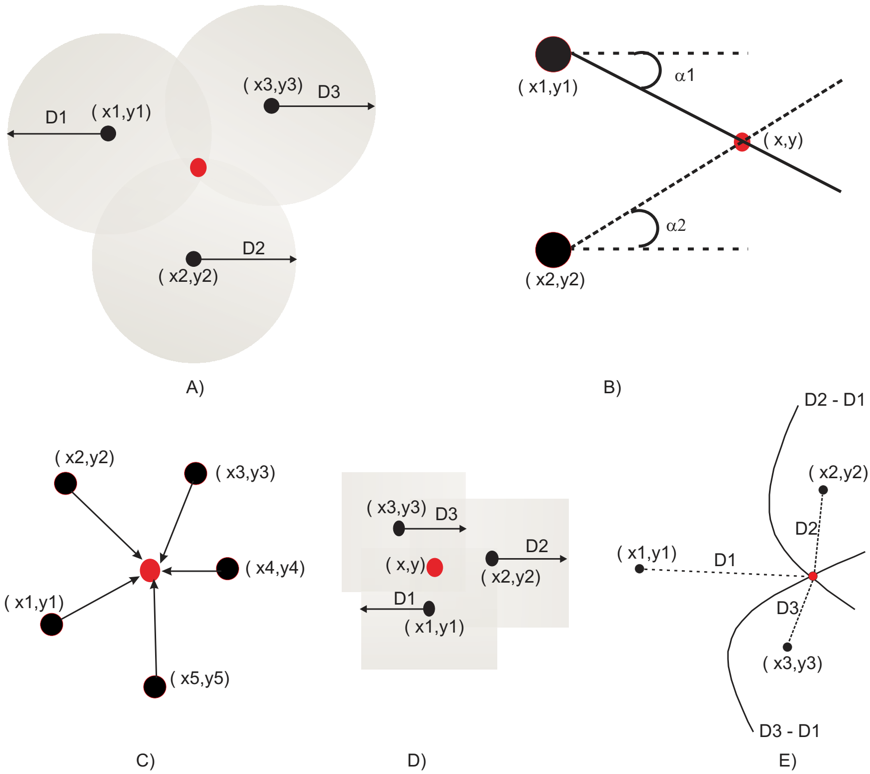


Figure 2.2: *Estimation methods: (A) Trilateration (B) Angulation (C) Multilateration (D) Min-max or Bounding box (E) Hyperbolic localisation.*

- **Proximity** measures the nearness (closeness) to a known set of points. The object's presence is determined using the physical phenomenon with limited range.
- **Centroid** estimates the geometric centroid of the beacon's location as the estimate of the object's location [44]. There are variants such as smooth centroid and weighted centroid algorithm [88].
- **Scene analysis** examines a view from a particular vantage point to draw conclusions about the observer's location [89, 109]. The scene itself can contain visual images,

such as frames captured by a wearable camera or any other measurable physical phenomena, such as electromagnetic characteristics that occur when an object is at a particular position and orientation.

- **Angulation** is a positioning procedure that relies on angle measurements with respect to the known beacons. At the minimum, two beacons are required to solve for a position–two beacons will result in four equations. If the equations are independent–that is, the mobile node does not lie on a line joining the two beacon nodes, the system is well-constrained and can be solved algebraically. To extend this method for 3D positioning, a third angle i.e. elevation is required from at least one of the beacons.

- **Lateration** uses distances to known reference beacons to estimate the location. It is popularly known as trilateration emphasising three distance estimates used to determine the location. The assumption here is to know the location of the reference beacons. The geometry of the reference beacons are important, as they provide enough constraints to solve the problem. For example, a collinear reference beacon arrangement will not allow for disambuigation of two solutions in two dimensions, regardless of the number of beacons.

- **Min-max** The main idea is to construct a bounding box for each beacon using its position and distance estimate, and then to determine the intersection of these boxes. The intersection of the bounding boxes is computed by taking the maximum of all coordinate minimums and the minimum of all maximums [118]. The estimated position by min-max is found to match closely with the position computed through lateration.

- **Hyperbolic localisation** For systems utilising TDOA measurements, intersection of hyperbolas will result in the final location estimate as opposed to intersection of spheres (lateration) and lines (angulation). The resulting solution might yield to multiple solutions. Optimisation algorithms discussed below could be used to solve for the optimal solution.

Optimisation or Error minimisation algorithms: Since measurements or observations are always associated with an unknown amount of errors, algorithms offering precise solutions with zero-valued residuals, a condition which is assumed by the above-mentioned algorithms, are always not valid. Optimisations or algorithms that minimise the error are designed to find solutions with the lowest possible residual for a given data. Basically, they iterate the solution space and compute expected measurements for each estimate of the solution. They require model equations that are used to predict measurements, and express the measured values in terms of unknown quantities. A common methodology here is to sum the squares of the nonlinear equations. This method is also known as *sum of squares* or *least squares optimisation*. They can be grouped into *gradient-based* or *stochastic-based* methods. Gradient methods involve use of derivatives to observe the rate at which an area of the solution space converges towards the optimum solution. The gradients are followed towards the error minimum. Some examples in this category are Method of steepest descent [159], Newton's method and Levenberg-Marquardt method [136]. Stochastic algorithms can be used for problems where the derivative of the error function is difficult or impossible to obtain. Examples include simulated annealing and particle swarm optimisation. Systems like Active Bat [188], self-calibrating tracking system from the University of Bristol [131], relate system [82] and audio location system [93] use some form of error minimisation techniques. The disadvantage of algorithms of this type is that they are computationally expensive [36]. Plus, the error minimisation algorithms do not make use of information obtained from previous position calculations.

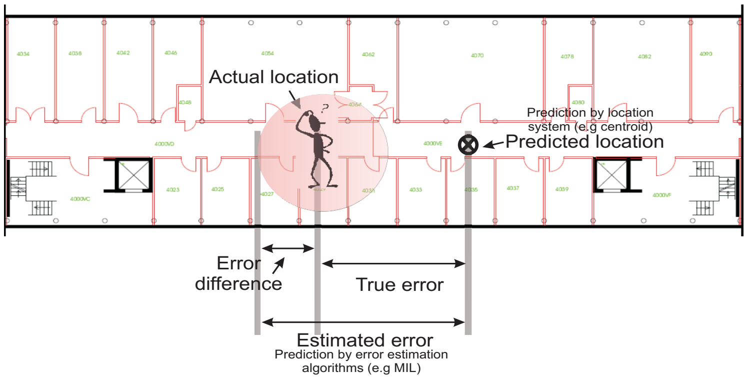
State-estimation algorithms: Algorithms that utilise the device's state (either current or current and past states) can be grouped under state-estimation algorithms. They operate by iteratively combining the previous estimate of the state with the observed measurements. They basically predict the current state using a model of process dynamics and the previous state and then correct the prediction using the current measurement. Many state estimation algorithms exist, of which Kalman filtering [193] is the most common. Variants of Kalman filtering and particle filtering are used as well. The Kalman filter is suitable for linear systems, however, most systems exhibit non-linearity. The Extended Kalman filter is developed for addressing this issue. It linearises the Kalman filter by applying a first-order Taylor series approximation to the process and measurement equations [144]. Particle filtering is an estimation technique that implements a recursive Bayesian filter using a Sequential Monte Carlo method. It is particularly good for dealing with non-linear and non-Gaussian estimation problems. It is based on a set of random samples with weights (or particles) for representing a probability density. It is often an alternative to the Extended Kalman filter (EKF) or Unscented Kalman filter (UKF) with the advantage that, with sufficient samples, it approaches the Bayesian optimal estimate, so they can be made more accurate than either the EKF or UKF. Welch et al. [193] gives an excellent overview of Kalman filtering and Hightower [88] presents a thorough comparison of various Bayesian filtering techniques for location systems.

2.6 Evaluation Criteria

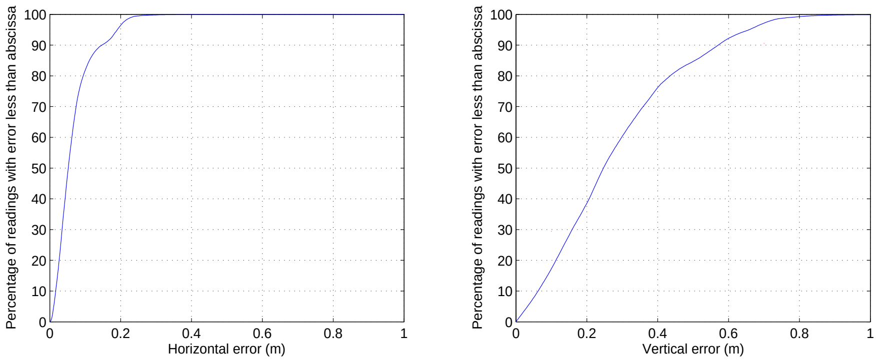
Location systems are evaluated based on the following criteria:

Accuracy and Precision: The key metric for evaluating a location system or algorithm is the *accuracy*. Accuracy is defined as, how much the estimated location deviates from the true location. The accuracy is denoted by an accuracy value and *precision* value (e.g. 18 cm accuracy over 95% of the time). The accuracy of a location sensing system is often used to determine whether the chosen system is applicable for a certain application. The precision indicates how often we expect to get at least the given accuracy. For example, WLAN and GSM localisation are applicable for wide spectrum of applications, but their accuracy has shown to be highly variable. In other words, the distance between the actual location and the predicted location fluctuates.

Typically quoted figures refer to either the root-mean-square (RMS) error of the system, or the median accuracy that 50% of readings will meet. However, neither of these measures give the system designer a good idea of what the outlying error distribution looks like, and it is frequently these errors that determine how effective a location system is in practice. A more reasonable way of describing the accuracy is through use of a cumulative probability graphs showing the fraction of readings having an error less than or equal to some value. A



(a) Accuracy



(b) Precision

Figure 2.3: *Errors in localisation (a) Accuracy and (b) Precision.*

typical cumulative probability graph (CDF) for an UWB system is shown in Figure 2.3 as an example. It can be seen that 90% of readings produced by the system lie within 18 cm of the true 2D position, and 90% lie within 60 cm vertically. Comparing data in this form is easier to assess the usability of the location system for a particular application. It is also important to consider whether a given location system is accurate enough for the application in question.

Hightower [89] suggests that a more complete representation of performance characterisation of any location system is based on error distributions with any relevant dependencies such as the density of infrastructural elements. As an example ,"Using five base stations per 300 square metres of indoor floor space, location sensing system X can accurately locate objects within error margins defined by a Gaussian distribution centered at the objects true location and a standard deviation of 2 metres". In addition to its comparison value, researchers can use a location-sensing system's accurately described error distributions as partial input for simulating a system.

It should be noted, however, that accuracy is only one of the many desirable location sensor properties and others, such as update rate and scalability are discussed below.

Geometric Dilution of Precision (GDOP): The quality of a reference beacon's geometry is described by geometric dilution of precision (GDOP). It is a measure of how the geometry of the beacons affects the propagation of errors in the measurements. This metric is appropriate for lateration and angulation based schemes. GDOP relates the fundamental ranging or bearing accuracy of the components of a location accuracy to its positioning accuracy at some point in space. GDOP typically magnifies the basic ranging and bearing measurement errors into the final accuracy of the location system. The location system should be installed in such a way to minimise GDOP to prevent ranging and angle errors from causing large position errors. A system with a good GDOP has a good spread of beacons along all the axes. For instance, collinear, coplanar arrangements need to be avoided.

Responsiveness or update rate: This metric specifies how often an estimate is produced. While this may not be important for devices that are static, it is of significant importance for devices in motion, as this can fully reproduce the motion of the device. Most of the motion tracking applications require the highest possible update rate to prevent motion sickness. Tracking humans does not, however, need to be at high update rate. On the other hand, only snapshots of the true path might be available from a system with a low update rate.

Scalability: Another important performance measure is the *scalability*. Scalability can mean two different aspects – one is the coverage of a particular location system being easily extended or scaled (e.g. by adding more rooms to an already covered area) and other aspect is to see if the system can cope well with an increasing number of locatables (or tags) that need to be located concurrently.

Cost: Cost can be split into costs that are associated with the setting up, installing, maintenance and operational cost. We divide the cost into the following categories:

- ***Infrastructure cost:*** This refers to the time and money required to deploy and maintain a location system. It very much relates to the coverage, although certain systems like GPS requires tremendous amount of installation cost, when we factor to the supported coverage, the cost becomes less.

- ***Tag/ client/ locatable cost:*** This refers to the actual tag/client/locatable cost that is required. It essentially refers to the incremental cost of adding more devices to be located by the system. Special purpose tags as locatables can turn out to be expensive while, personal devices acting as locatable can be extremely cost-effective.

- ***Calibration cost:*** Knowledge about beacons and sensor imperfections can be used to improve the accuracy of the location estimation. Device calibration is the process of forcing a device to confirm to a given input/output mapping. Almost all the location systems require some form of calibration.

- ***Power consumption:*** An important cost factor when running the system in a real environment is *power consumption*. When scaling to thousands or millions of autonomous small devices it is clearly not feasible to change or recharge batteries very often, thus *energy efficiency* should be a goal of any localisation mechanism meant for a large scale system.

Privacy: An important parameter of localisation system is *privacy*, which should form part of the architecture since its conception. Using localisation, it is very easy to create a Big Brother infrastructure that tracks user movements and allows to deduce patterns of behavior. This issue is being generally overlooked in the design of systems and considered as an after thought only. Tracking or centralised systems are particularly weak with regard to privacy.

2.7 State of the art

This section provides the relevant state of the art in location systems most of them developed after the advent of ubiquitous computing. They are categorised based on the type of technology they use to determine location. The field of localisation is active and vibrant, with research in localisation continuously being developed, and market players are constantly growing. For this reason the examples presented in this survey are not exhaustive, but are carefully chosen to cover a wide variety of technologies that we have discussed in the previous sections.

2.7.1 Location Systems

Satellite-based Loran was the first navigation system, launched before the World War II to employ TDOA based radio signals [152]. It was also the first true 2-D position finding system. Transit was the first operational satellite based navigational system launched in the year 1959 and was still in use until the early 1990s, when it was made obsolete by global positioning systems (GPS). Transit users determine their position by measuring the Doppler shift of signals transmitted by the satellites. Transit had several shortcomings–to avoid radio interference, the system was limited to using five operating satellites simultaneously which resulted in temporal coverage with periods of unavailability. Secondly, the system produced only 2-D positions it was not usable for aircraft navigation applications.

GPS [152] is one of the oldest location technologies that provided the location of the users in 3D. GPS is a one-way ranging system where all signals are transmitted by earth-orbiting satellites and position estimation happens locally at the receiving units. GPS system architecture consists of three parts: a constellation of earth-orbiting satellites that broadcast a ranging signal, ground stations that update the satellites coordinate projections and clocks and receiver units that use the GPS signals to estimate their position. This design makes it possible for GPS to provide worldwide coverage and to scale to an unlimited number of users, while at the same time preserving user privacy [120]. Accurate GPS positioning requires an unobstructed view of at least four satellites. GPS signals do not penetrate well through walls, soil, prohibiting its usage for positioning inside buildings, underground (e.g. mine or tunnel). GPS signals can also be obstructed by large buildings that are typical to urban environments. Table 2.3 shows the comparison of various outdoor geo-location systems.

GPS has now become integrated in day-to-day navigation–most of the new cars are sold with a GPS navigation system and personal devices such as cellular phones are increasingly being integrated with GPS chipsets. The market for GPS products and services was estimated

System	Method	Coverage	Dimensions	Accuracy
Loran	Hyperbolic	Cont	2D	250 m
Transit	Doppler shift	Global	2D	25 m
GPS	Spherical	Global/Cont	3D	5-10 m

Table 2.3: *A comparison of several outdoor geo-location methods based on satellites.*

at €15 billion in 2001 and is expected to grow to €40 billion by 2015 [167]. It is estimated that by the year 2020, the number of GPS chipsets will approach 3 *billion* [97]. Commercial players include companies like Garmin [10] and Tom Tom [20].

Infrared-based systems The *Active Badge* [186] is one of the early centralised indoor personal location system making use of infrared (IR) technology. Each person in the office wears a badge, which emits a unique IR signal that is then gathered by the network of sensors and collected by a master server. The information is then relayed to the visual display unit. This was mainly used as an aid for the telephone receptionists to direct the phone calls to appropriate persons during working time. Inherent to IR are the limited range and inability to pass through obstacles like walls and jackets. Nevertheless it works well for the intended application of routing telephone calls. A commercial infrared badge, the Versus Personnel Alert Badge is marketed to hospitals to track hospital staffs and equipment [185].

Locust Swarm location systems from MI [100] uses solar powered units affixed on the ceiling near light sources broadcasting a unique code associated with the surrounding area. Mobile users carry wearable computers to infer their location from the received infrared codes. Granularity is limited to the space in which the locusts are deployed.

Krohn et al. describe a small scale system to determine position and orientation of nearby personal appliances (laptops, PDAs, etc.) towards each other [106]. The system does not rely on any pre-installed infrastructure. Instead, one device is chosen as the origin of an arbitrary coordinate system. Distances and angles between devices are determined. Two strategies are presented to infer locations, a non-linear regression algorithm being the more robust one. Using this algorithm and four devices placed in an area of $160\times200\ cm^2$, 95% of the location estimates are accurate to 10 *cm* in position and 14° in orientation.

FoxTrax System [47] was designed to track an ice hockey puck and to enable its position to be highlighted on screen during the televised ice hockey game. The puck emits infrared signals that are detected by the field of view of infrared cameras. The system requires guaranteed line of sight. In a relatively open environment of a hockey stadium, a combination of 20 IR emitted diodes in the puck, 20 detectors in the ceiling rafts and 10 IR cameras were required in the FoxTrax system to ensure continuous line of sight to several detectors and several IR cameras simultaneously.

Ultrasound-based systems Compared with radio and infrared-based system, fine-grained ultrasonic location systems typically require dense network of sensors (typically called transducers) to be installed in the environment. In *Active Bat* [80], the bat is attached to the objects or persons whose location has to be determined. These bat transmitters emit ultrasound pulses, which are received by the receiver mounted on the ceiling. A central controller coordinates the transmitters and receivers. To locate a particular bat, the controller sends a unique ID over the radio channel. When a bat detects its ID, it sends an ultrasound pulse, which is picked by receivers in the ceiling. From the time-of-flight measurements, the system can calculate the 3D position of a bat with an accuracy upto 3 cm. Since the Active Bat system uses the time-of-flight information from ultrasonic pulses to determine the location of a bat, it is vulnerable to indoor multipath and reflection effects from walls and other obstacles. The Bat system has a degree of built-in robustness to mitigate these effects, as at least three bat receivers must agree on a location using their multi-lateration algorithms (based on a non-linear regression model) before a measurement is considered as valid. Ward [189] proposes methods to eliminate measurements due to reflections and multipath. The system is reported to work well in the middle of the rooms, it breaks down slightly when a bat is placed very close to a large obstacle such as a wall.

Cricket [160] makes use of proximity-based lateration techniques for providing location information. Many beacons installed at known locations advertise the identity of that space with the use of some character string. Every device in the network has a listener attached to it. The listeners use some inference algorithm to determine the space in which they are currently located by listening to the beacon announcements. Each beacon sends two signals, an RF signal carrying the location data and an ultrasound carrying a narrow pulse. Based on the difference of arrival times, the device finds the absolute distance between the beacon and the listener. In a similar approach, Randell et al. [163] uses four ultrasonic transducers placed at the corners of a square on the ceiling and wired to a controller. The controller sends an RF trigger, and then issues a pulse from each of the four transmitters in succession. A mobile receiver unit connected to a handheld computer receives the pulses, and estimates its location with accuracies between 10 and 25 cm. The Cricket system uses a combination of US with RF to solve the problem of time synchronisation. This is accomplished by exploiting the difference in the propagation speed between an RF pulse and an US pulse that are sent simultaneously. The difference in TOF between the RF signal and the US signal is used at the client node to calculate the distance to the beacon node. Note that the previously explained Active Bat system uses US exclusively, and solves the time synchronisation problem by using cables between the infrastructural nodes.

The Constellation system [68] tracks a mobile unit consisting of a 3D inertial sensor and a number of ultrasonic sensors. Location is calculated using TOF measurements between the mobile unit and fixed transmitters in the environment. After the initial starting position the inertial sensors measure the location of the users current location which are periodically corrected by making use of ultrasonic TOF measurements. An accuracy of approximately 5 mm is reported, but the mobile tracking unit, worn on the head and belt of a user, is too obtrusive and expensive.

A system developed by Hazas and Ward [84] extends ultrasonic capabilities by using broadband signals. This has significant advantages over the presented narrowband ultrasound transducers for ranging in terms of providing high update rates, reducing the signal interference.

Radio Frequency-based The earliest indoor location systems, such as the Active Badge, Active Bat introduced new infrastructure to support localisation [80, 186]. Despite some success, as indicated by commercialised products [182], the cost and effort of installation is a major drawback to wide-scale deployment. New projects in location-based systems research reuse existing infrastructure to ease the burden of deployment and lower the cost. The earliest demonstrations use 802.11 access points [33, 115], and more recent examples explore Bluetooth [130] and wireless telephony infrastructure, such as GSM [151] or FM transmission towers [108].

- WLAN: *RADAR* [33] makes use of RF signals for finding the user location indoors. It adopts radio fingerprinting to build a radio map of the building. A mobile device takes a series of measurements and records signal strength of a single source as reading. The accuracy depends on number of measurements points (experiments have revealed measurement on every square meter on average is required). Then a centralised system gathers signal strength information from multiple receivers and estimates location by comparing the measured signal strength with the signal strength that were gathered in the training phase, that is to say choosing the location of the training point with the closest Euclidean distance in the signal space.

  Ekahau Positioning system is a software based positioning solution that can continuously pinpoint and track the location of mobile computing devices with a reported accuracy of 1–2 m in indoor and campus environments. Ekahau technology does not require any additional wireless infrastructure on top of the standard WiFi network [56].

- GSM:

  The cellular-based location systems are broadly categorised into four types–*cell-id based*, *assisted-GPS*, *signal strength modeling* (can be time-based, angle-based or signal strength-based) and *radio fingerprinting*. It is also possible for base stations to perform round-trip timing for distance measurement, but this is currently not made available to the handsets. Assisted GPS is a system intended for use with mobile phones [53]. The telephone handset contains a simplied version of a GPS receiver. Nearby cellular base stations supply the handset with information on the current satellite signal conditions. This allows the handset to improve signal-to-noise ratio, and to reduce the time required to resolve satellite signals. An assisted GPS handset is accurate to 15 m outdoors and 50 m indoors. A fingerprinting approach by collecting wide-area fingerprints of GSM signal is researched by Otsason et al. [151]. Using more measurements usually provides more accurate results; for example, incase of the GSM based positioning system developed by [151] RSSI measurements of up to 40 GSM cells are used. Location systems based on cell-id, modeling approach and assisted-GPS are commercialised. However, the fingerprinting-based approaches are still under research.

- Bluetooth and others: Bluetooth location systems, using timing of RF signals between cooperating nodes, are being commercialised by companies like Bluesoft [6]. Some modifications to the Bluetooth radio and protocols are required to implement this functionality.

Pinpoint 3D-iD [157] requires proprietary base stations and tags are used to measure radio time-of-flight. It uses an installed array of antennas at known positions to perform multilateration. Pinpoint's accuracy is roughly 1–3 m. The Pinpoint system measures the round-trip TOF of a spread spectrum radio signal sent in the ISM band from a fixed infrastructure of transceivers placed around a building to active transponders placed on equipment.

In the *SpotOn system* [87], special tags use radio signal attenuation to estimate the distance between tags. The aim here is to localise wireless device relative to one another, rather than to fixed base stations, thus allowing ad hoc localisation.

- RFID:

  RFID-based systems use RFID readers, RFID tags and communication between them. One example is the *Landmarc* systems [145]. Passive RFID systems infer the presence of nearby objects using a small passive transponder tag attached to those objects. A reader unit energises transponders in its vicinity via electromagnetic coupling, causing the tags to respond with a radio message containing their unique identifier. The tags are cheap and virtually indestructible but they need to be placed in close proximity of the reader antenna to be detected. Active RFID systems use battery powered tags which transmit signals coded with tags unique identifier. Since they do not rely on electromagnetic coupling to provide power to the tag, they operate over much longer ranges, but their tags are more expensive.

  Aeroscout [2] system provides location using an 802.11 architecture by measuring the time-of-arrival of packets from a set of active RFID tags to a set of location receivers. The advantage of the WiFi-based RFID is that the tags can be detected by commercially available wireless access points. The system consists of a set of active RFID tags and a number of specialised location receivers (long-range RFID readers). Location is estimated using a combination of time-of-flight based triangulation and RSSI and reports an accuracy of 1–5m. WhereNet [198] is location system operating in the 2.4 GHz band. The active RFID tags are attached to objects to be located, and their signals are detected by a set of receivers placed at known points.

Electromagnetic-based Polhemus Inc [16] and Ascension Technology Corp [4] are two existing producers of state-of-the-art electromagnetic tracking systems. Many of their products have sub-centimetre position accuracy and sub-degree orientation accuracy. The system operates over an area of 57 m^2 and calibrated accuracy over this area is said to be 5 *cm* in position and 3° in orientation. A maximum of 120 measurements per sensor can be made per second. MotionStar by Ascension Technology [4] detect the position based on the same principle and offers good accuracy and update rates. However, they are expensive and have high power consumption and are sensitive to the presence of metallic objects in the environment.

Ultra-wideband-based Ubisense system entities comprise of ubisensors (receivers) and ubitags (transmitting UWB pulse) at a peak update rate of approximately 10 Hz. *Ubisense* [176] uses a combination of TDOA and AOA to locate people and objects with an accuracy of upto 15 *cm*. Sensors mounted in the area to be monitored tracks Ubitags attached to objects or carried by people are then automatically tracked to provide accuracy and reliability. UWB cope well with the multipath fading. This is because by using short pulse length the direct path can be discriminated from echos more easily. A leading edge detector performs this discrimination and is presented by Fontana [64]. A commercial version of their system based on TDOA is available from Multi-Spectral Solution (MSSI).

Optical (vision)-based Using cameras to track the location of the people are increasingly used both indoors and outdoors for security and surveillance [50]. Fixed cameras are also used in smart environments to track people without the usage of tags or badges. One such example is Microsoft's *Easy Living* [109] which uses Digiclops real-time 3D cameras to provide stereovision-positioning capability in a home environment. The EasyLiving Person Tracker is limited to tracking two or three subjects in the cameras field of view, and returning location estimates for them several times per second. The significant drawback is that the system is not readily scalable in terms of the number of entities which can be tracked in a given area. If the number of people increases within the field of view, it increases the scene dynamics and make occlusions more frequent. There are also vision-based systems using markers. The reacTIVision system [35] is software for tracking specially designed fiducials (markers) in a real-time video stream. The system uses computation to minimise marker size while meeting geometric constraints required to compute the location and 2D orientation of the markers. The tag must be within the field of view of the camera for the system to work precisely. TRIP (Target Recognition using Image Processing) has been developed by Lopez de Ipina [52]. Users wear passive tags displaying 2D circular bar codes. Cameras in each room capture images which are analysed to identify tag wearers in the field of view. Provided a tag is within 3 m of a camera and is turned away from the camera at an angle of no more than 70°, its position can be computed with an accuracy of 10 *cm*. A single PC processing images from a single camera can achieve an update rate of 16 *Hz*, even with many targets in the field of view. In addition to the position, the pitch and yaw of the target with respect to the camera can also be estimated.

Load-based Georgia Institute of Technology's *Smart floor* [150] identifies people based on their footsteps. However, this technologies negative side is the huge installation cost and infrastructure cost. An example is the Active Floor [27]. Based on sensing pressure by analysing the weight distribution across the floor, moving objects can be located. The authors use Hidden Markov Model to analyse footstep patterns. Although load-sensing floors are able to identify moving subjects reliably, their ability to track subjects on the same floor area has not been thoroughly tested. Also, tracking is limited to subjects who walk or move along the ground; devices of interest (such as PDAs, cameras, and computers) cannot be easily tracked. Thus, like vision-based systems, load-sensing floors could face scalability issues when confronted with many subjects in the same area.

Inertial-based Inertial systems have gained a lot of attention lately, with many products emerging, XSens [199], pi-node from Philips [156] are just a few examples. The positions provided by inertial sensors unavoidably drift over time due to errors in measurements being integrated [66]. Despite the limitations dead reckoning is the only completely self-contained location technique that requires no prior knowledge of the environment. The drift can be reduced by using shoe-mounted inertial sensors and resetting the velocity to zero at each footfall [149] and by combining the inertial measurements with data from an electronic compass through a Kalman filter in order to avoid drift in the heading [67]. In most cases it is essential to correct positions and headings with data from external sources. GPS is one possibility but only for outdoor navigation with short periods of GPS outage [165]. Another possibility is to predeploy RFID tags at known locations and use these to correct positions [201]. Indoor location systems such as Ubisense have also been used in combination with PDR [86].

The navigation system developed by Renaudin et al. [166] for emergency responders combines PDR with map matching in order to prevent drift. Inertial measurement units (IMUs) on the chest and legs are used to measure movement and posture. The first rescue team to enter the building places an RFID tag on each door frame they pass through. The position computed by the inertial navigation system (INS) can then be corrected according to a database of the coordinates and directions of all doors in the building. The second team is equipped with an RFID reader and can therefore determine its positions by scanning for each tag. Apart from providing dead reckoning solutions, inertial systems have been used in autonomous vehicles [38], where inertial sensors are periodically calibrated by periodic stopping.

Table 2.4 gives a global view of location technologies classified by achievable accuracy and the possible applications. The list of applications are provided to serve as an example.

Technology	**Accuracy**	**Example**	**Possible Applications**
Satellite	5-10 m	GPS	outdoor navigation (land, sea, air), pet tracking, tour guides
Cellular	5-50 m	A-GPS	emergency response, social networking
Infrared	5-10 m	Active Badge	asset/personnel tracking indoor navigation tour guides nearest printer, teleporting system
Ultrasound	1-10 cm	Active Bat	tangible user interfaces, fine-grained services, asset tracking, walk through video phone
Vision	1 cm-1 m	EasyLiving	security, surveillance, elderly care
UWB	6-10 cm	Ubisense	asset and personnel tracking indoor navigation emergency response etc.
Bluetooth	2-10 m	BlueSoft	proximity detection, navigation
WLAN	2-100 m	Radar	indoor navigation, tour/museum guides, social networking
RFID	5 cm-5 m	LandMarc	asset and personnel tracking indoor navigation emergency response, logistics
EM	5 mm–5 cm	MotionStar	motion capture

Table 2.4: *Summary of existing localisation systems. Accuracy as reported in [83].*

2.7.2 Algorithms for sensor network localisation

Wireless Sensor Networks (WSN) [28] have appeared as one of the emerging technologies that combine automated sensing, embedded computing and wireless networking into tiny embedded devices. Although these individual enablers of WSNs are themselves not new ideas, technological improvements, particularly in micro-electro-mechanical systems (MEMS), enabled their integration [58] on miniaturised embedded computers that support the concept of the disappearing computer. *Location* and *orientation* information of objects in such networks is useful for both services and applications. *Services* that are enabled by availability of location, includes routing [122]–(geographic assistance in ad hoc routing promises significant reductions in energy consumption), resource management , service discovery and querying– query nodes over a specific geographic area. Location can also be used to study the coverage properties of a sensor network [133]. At *application level*, location is required in order to label the reported data in a sensor network without which gathered data is meaningless, whereas position together with *velocity* and *orientation* enable tracking. Langendoen et al. [117] reports about the 150 sensors nodes deployment in a potato field to measure the microclimate as part of a precision agriculture experiment. It is very time consuming to manually determine the locations of all the nodes and is desirable for them to discover their own position. An unmanned aerial vehicle (UAV) is used to drop sensors nodes which then detect and track passing vehicles [158]. Since the nodes are dropped from a height, only their very approximate location is known. Each node must determine its exact position. In both these applications knowing the location of the nodes is not the primary goal but without this information the rest of the data is useless.

Some of the important properties needed for sensor node localisation are distributed algorithms making use of connectivity information (as they are typically deployed in large numbers), self-organisation and computationally efficient. We describe range-free algorithms that are used in most of the sensor network research in this section.

Range-free schemes make no assumption about validity of distance or angle information like the range based schemes. Some examples to quote are *Centroid algorithm, APIT, amorphous localisation* and *DV-Hop algorithm*. In the centroid method [43], each node estimates its location by calculating the center of the locations of all anchors (or beacons) it hears. If anchors are well positioned, the location error can be reduced [45], but this is not possible in ad hoc deployments. The APIT method [85] isolates the environment into triangular regions between beaconing nodes and uses a grid algorithm to calculate the maximum area in which a node will likely reside. DV-based positioning algorithms are localised, distributed, hop-by-hop positioning algorithms [146, 147]. They work as an extension of both distance vector routing and GPS positioning in order to provide approximate positions for all nodes in a network where only a limited fraction of nodes have self positioning capabilities. They use the same principle as of GPS, with the difference that the landmarks are contacted in hop-by-hop fashion rather than a direct connection and similar to distance vector each node at any time can communicate only with its neighbors. The amorphous method [142] is similar to DV-hop as the coordinates of the anchors are flooded throughout the network so each node can maintain a hop count to that seed. Nodes calculate their position based on the received anchor locations and corresponding hop count.

An often ignored issue in ongoing research is the impact of beacon density and the placement of the beacons. Self-configuring localisation systems consider beacon density as an important parameter in characterising the localisation quality. HEAP and STROBE [44] are dependent on density of beacons in the network.

In the *convex optimisation* [54] approach, the positional information is inferred from connectivity imposed proximity constraints. Few nodes have known locations, called the anchor nodes, and the remaining nodes infer their position from the knowledge about communication links. *MDS-MAP* [171] is a method that makes use of connectivity information to provide locations in a network with or without beacons (known co-ordinates). The advantage of MDS-MAP is that it has a wide range of applicability due to its ability to work with both simple connectivity and range measurements to provide both absolute and relative positioning [171, 146]. Both convex optimisation and MDS-MAP require centralised computation. Recently research on localisation is focused on incorporating the mobility model. Lingxuan Hu et al. [92] use a *sequential Monte Carlo Localisation* method and argue by exploiting mobility accuracy and precision of localisation are improved. Probabilistic techniques, such as *Markov modeling, Kalman filtering* and *Bayesian analysis* can also be used to determine the absolute location of a mobile node [112]. Fingerprinting-based methods are also applied to sensor network localisation [128].

2.8 Rationalisation for multimodal localisation

It is evident from our presented survey that the current localisation landscape is an amalgamation of location systems based on a multitude of different technologies. Despite the plethora of established location technology, there is no single location technology that may be relied upon in all environments to provide accurate location information. Clearly *"no one size fits all"* and a pervasive location system is not yet available.

In Table 2.5, we list the most prominent existing systems and how they fit in our earlier defined taxonomy. Apart from the criterions that we mentioned in our taxonomy, we include a column called "fusion" in Table 2.5. This essentially refers to whether there are more than one type of information used by that specific system to compute location. From the list of surveyed systems in this chapter, it is evident that location systems that employ *only one form of sensing* all suffer inherent drawbacks. For example, pure inertial sensors suffer from drift [66], ultrasound sensors require clear line of sight, and magnetic sensors are affected by ferromagnetic and conductive materials in the environment and GPS systems requires a certain amount of time to get fixed to the satellites. Figure 2.4 gives the performance characteristics of the state-of-the-art location technologies that we have described thus far. As one may note the accuracy of the location systems are well correlated with their deployment cost– ranging from easy-to-deploy coarse-grained systems to expensive, carefully tuned, calibrated, fine-grained systems.

Fusion typically refers to the effective use of two or more heterogeneous sensor observations to determine location. The primary aim of using data fusion is to improve the quality of the location estimates and identity of entities or to make inferences that may not be feasible from a single sensor alone. Typical benefits include [78]–(i) robust operational performance, (ii) extended spatial and temporal coverage, (iii) increased confidence, (iv) reduced ambiguity, (v) enhanced spatial resolution, (vi) improved system reliability, (vii) increase in update rate and (viii) reduce effects of errors in measurement.

Inertial sensors' inherent drift can be corrected by adding an external source of information. For instance, Foxlin demonstrated a system that is inertial based but aided by ultrasound sensors that can make small adjustments in position and orientation [68]. GPSs' time-to-fix problem can solved by A-GPS systems which relieves the load of computation to servers thus, making it faster in getting the position fix. While accuracy can be one key factor of improvement, there are other factors, like reduction in number of beacons employed. For instance UWB-based Ubisense system [182] incorporates both AOA and TDOA measurements, thereby reducing the density of beacons to be deployed for localisation. Fusing mul-

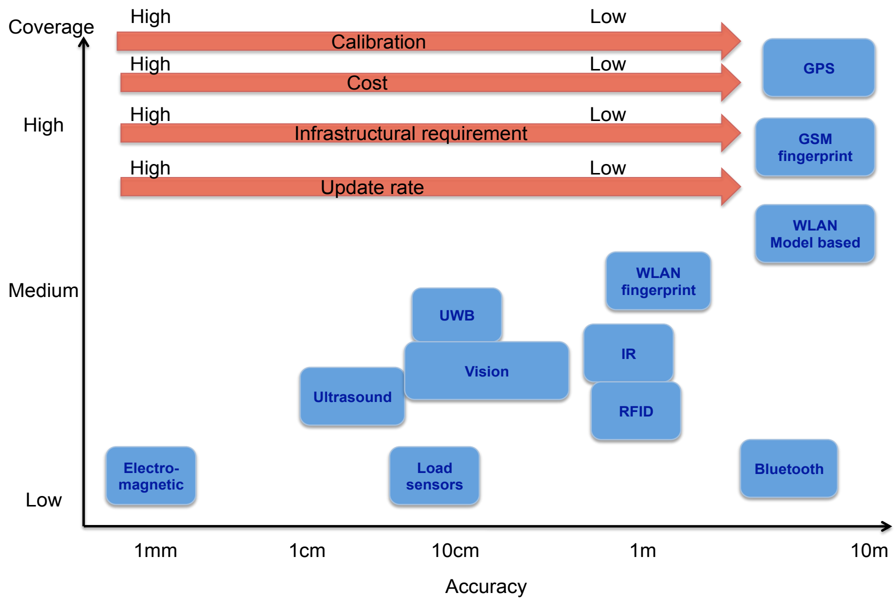


Figure 2.4: *Performance characteristics of the state-of-the-art location technologies based on what we have reviewed in Section 2.7 illustrating, how accuracy, coverage and cost does not co-exists.*

tiple technologies like US together with RF can solve the time synchronisation problem, for instance Cricket systems [160] uses the difference in propagation speed between an RF pulse and an US pulse to calculate the distance to the beacon node.

Apart from improving performance of the location system in limited measurement volumes, fusion of heterogeneous sensing systems will ultimately allow people to move from room to room in a building without loss of location knowledge and thereby truly enabling the vision of Mark Weiser. This type of seamless operation between different areas has been addressed in Hightower's work [88] and has made a significant impact on the field including commercial adoption by Intel [74], research adoption by the Placelab project [12], and community adoption through the publicly available location estimation library. Intel's Universal Location Framework (ULF) [74] provides mobile users with a seamless hand-off between location services as they move their client devices between indoor and outdoor environments. Specifically, it fuses readings from a GPS receiver outdoors with WLAN RSSI-based triangulation when indoors. The process of blending different data is referred to as *multimodal data fusion* and is the focus of this thesis.

The particular quantitative improvement in estimation that results from using multiple sensors depends, of course, on the performance of the specific sensors involved (data collection rates, observational accuracy), environmental effects, and the specific algorithms used in the data fusion estimation process. Thus, it is very important to have the knowledge about the sensor data we are dealing with to see if the fusion do result in improvement. Hence one of the main focuses of this work is characterising the measurements/technologies that are most commonly used for localisation purposes. One of the other merits of characterisation is that the positioning algorithms by themselves require to feed in error distributions to function effectively. We detail this important step all through Chapters 4 – 6.

The primary goal of the thesis is to address the benefits of multimodal localisation, with a specific focus on improving accuracy. We illustrate various ways to achieve this goal (on a range of technologies) ranging from simple smoothing/averaging of multiple location readings to sophisticated fusion and tracking, which has the capability to combine data from heterogeneous sensing modalities to improve the quality of the location estimates.

Specifically, in Chapter 4 we demonstrate the effect of merging information that is obtained from the same source, i.e. WLAN RSSI is used to infer *motion* and used for improving the *location* accuracy. This could be considered as smoothing of location estimates based on motion derived from simple and easily available RSSI and as a result eliminates the so-called "teleportation effect" that commonly occur in location algorithms using RSSI data. We demonstrate how simple algorithms like Centroid and Weighted centroid, could improve performance from knowing the motion of the device to be located and by using history of past location readings.

While smoothing of location estimates can be one easy way to improve the quality of the final estimate, *fusion* and *tracking* are sophisticated ways to improve the accuracy. The algorithms we describe in Chapter 4 are insufficient to handle these tasks. The goals of Chapter 5 and 6 are to demonstrate the benefits of fusion and tracking on sophisticated data such as the TOA, AOA and TDOA measurements. Tracking has the capability to provide a continuous stream of location estimates, even amidst the absence of input observations. Chapter 5 addresses the benefit of heterogeneous observations (pseudoranges and angles) gathered from an ultra-wideband system. We specifically focus on algorithms based on error minimisation and state estimation approach. In Chapter 6, we address the usage of ultrasound and inertial sensing technology to provide navigation and tracking solutions. We focus on correcting the drift in inertial sensors by deploying ultrasound sensors as landmarks and fusing the two complementary technologies using algorithms based on a state estimation approach.

Table 2.5: *Comparison of location systems, based on taxonomy defined in Section 2.2. The last column "fusion" highlights what specific benefits fusion brings in.*

Systems/ Authors	**Dependence**	**Coverage**	**Range**	**Technology**	**Measurement type**	**Estimation method**	**Architecture**	**Computation**	**Output representation**	**Object association**	**Fusion**
GPS [152]	Inf-based	Outdoors	Long	Satellite	TDOA	Lateration	Passive	Localised	Phy/Abs	Tagged	No
A-GPS [53]	Inf-based	Out/In-doors	Long	Satellite + cellular	TDOA	Lateration	Passive	Localised	Phy/Abs	Tagged	Yes (Satellite + Cellular = reduced time to fix)
GSM-modeling [184]	Inf-based	Outdoors	Long	Cellular	TDOA	Lateration	Passive	Localised	Phy/Abs	Tagged	No
GSM-FP [151]	Inf-based	Outdoors	Long	Cellular	RSSI	Finger printing	Passive	Localised	Phy/Abs	Tagged	No
Ekahau [56]	Inf-based	In/Out-doors	Medium	WLAN	RSSI	Finger printing	Active/ Passive	Cent./ Localised	Phy/Sym	Tagged	No
Radar [33]	Inf-based	In/Out-doors	Medium	WLAN	RSSI	Finger printing lateration	Active/ Passive	Centralised	Phy/Sym	Tagged	No
Placelab [115]	Inf-based	Outdoors	Medium & Long	WLAN, Cellular Bluetooth (BT)	RSSI	Centroid Lateration	Passive	Localised	Phy/Sym	Tagged	Yes (WLAN + Cellular + BT = increased coverage)
Aeroscout [2]	Inf-based	Out/In-doors	Medium	WLAN	RSSI + TDOA	n/a	Active	Centralised	Phy/Sym	Tagged	Yes (WLAN + RFID tags = reduced tag costs)
SpotOn [87]	Ad hoc	Indoors	Medium	RF	RSSI	Lateration	Passive	Distributed	Abs/Sym	Tagged	No
Active Badge [186]	Inf- based	Indoors	Short	IR	Proximity	Proximity	Active	Centralised	Sym/Abs	Tagged	No

Continued on the next page

Table 2.5 Comparison of location systems, based on taxonomy defined in Section 2.2.

Continued from previous page

Systems/ Authors	Depend-ence	Coverage	Range	Techno-logy	Measure-ment type	Estimat-ion method	Architect-ure	Compu-tation	Output represent ation	Object association	Fusion
Krohn et al. [106]	Ad hoc	Indoors	Short	IR	Range/ angles	NLR	n/a	Distributed	Rel	Tagged	No
Active Bat [80]	Inf-based	Indoors	Short	US	TOF	NLR lateration	Active	Centralised	Phy/Sym	Tagged	No
Cricket [160]	Inf-based	Indoors	Short	US+RF	TOF	Lateration	Passive	Distributed	Phy/Sym	Tagged	Yes (RF + US = resolves time sync.)
Constellation [68]	Inf-based	Indoors	Short	US + inertial	TOF	Kalman Filter	Passive	Centralised	Phy/Sym	Tagged	Yes (US + inertial = improved accu-racy)
MSSI [64]	Inf-based	In/Out doors	Medium	UWB	TDOA	Proprie-tary	Active	Centralised	Phy/Sym	Tagged	No
Ubisense [182]	Inf-based	In/Outdoors	Long	UWB	TDOA + AOA	Proprie-tary	Active	Centralised	Phy/Sym	Tagged	Yes (TDOA + AOA = reduced bea-con density, increased robustness)
Polheums [16]	Inf-based	In/Outdoors (except mag-netic interfern.)	Short	EM	Pulsed AC	n/a	Active	n/a	Phy/Abs	Tagged	No
Active Floor [80]	Inf-based	Indoors	Short	Load sensing	Load sensing	HMM	n/a	n/a	Phy/Abs	n/a	No
TRIP [52]	Inf-based	Indoors	Short	Proximity	Proximity	n/a	Active	Centralised	Phy/Sym	Tagged	No
EasyLiving [109]	Inf-based	Indoors	Short	Vision	Feature extraction	n/a	n/a	Centralised	n/a	Untagged	No

CHAPTER III

Application Settings

We explore localisation algorithms that use multiple sensor modalities to bring performance benefits. To ground our work, we have chosen three specific applications. In this chapter we show how the presented taxonomy (Chapter 2) can be used to assess the suitability of an existing technology for the chosen application. Our taxonomy lets us create a design plan of a location sensing system that meets the needs of an application. It essentially helps in ruling out certain options and analysing the suitability of the others. For each of the motivating applications that we present in this chapter, we first outline a problem statement justifying why positioning is needed and what are the existing solutions and outline the requirements.

1. Localisation in office environments to facilitate social networking, as a way to help coordination of people and understand social patterns. We leverage the existing wireless local-area networks (WLAN) infrastructure to sense motion and location with the main motivation of building wide-area location services.

2. Transport and logistics operation (e.g. in warehouses), motivating the need for fine-grained location information. We use ultra-wideband (UWB) as it copes with harsh indoor environments better than conventional radio technologies.

3. Emergency response scenarios, motivating the need for ad hoc positioning capabilities. In particular, we use a combination of inertial sensors and ultrasound sensors. The position error in a purely inertial system increases with time and requires correction from external sources. We address this problem by deploying ultrasound sensors as landmarks correcting for the inertial drift.

3.1 Application I – Location and tracking in office environments

Location technology has been explored as an enabling technology for social networking, as a way to help people coordination and understand social patterns [5, 9]. In this section, we briefly discuss the scenario, requirements and technologies for enabling indoor localisation in office spaces. Typical applications within office environments could range from tracking documents, providing navigational guidance to finding colleagues. These applications could also be applicable to large conference arenas or exhibitions (or trade-fair) such as CeBIT [7].

Figure 3.1: *Active Campus [1] includes both a buddy list tool with enhanced location information (left) and a map-based social awareness tool (shown right).*

3.1.1 Existing approaches and trends

Location systems that provide fine-grained information, such as the Bat systems [80] or Vision-based systems [109] are typically operational within the confined area of deployment. Recently there has been a growing interest in providing *wide-area location services*, that spans larger area (e.g. city-wide or campus-wide). Projects such as Active Campus [1](shown in Figure 3.1) aims to provide location-based services for educational networks and understand how such systems are used. For instance, activeclass enables collaboration between students and professors by serving as a visual moderator for classroom interaction. ActiveCampus Explorer uses a persons' context, like location, to help engage them in campus life. This trend has been demonstrated by some of the key commercial companies involvement such as Intels' Placelab [12], Intels' ULF [74], Skyhooks' Wireless positioning services [17] and more recently, Googles' Latitude [72]. The main motivation for building wide-area location services is that many of the state-of-the-art systems (that we discussed in Chapter 2) coverage is limited to a particular room or building. Applications like location-aware instant messaging would have to fall out if they have to be only operational within such a limited working area.

Placelab [12] focuses on addressing this specific issue of *maximising coverage* and providing a low barrier to entry for *users and developers*. The Placelab approach is to allow commodity hardware clients like notebooks, PDAs and cell phones to locate themselves by listening for radio beacons such as 802.11 access points (APs), GSM cell phone towers, and fixed Bluetooth devices that already exist in the environment. Placelab has even more ambitious goals by seeking to create a comprehensive location database which uses fixed commodity WiFi, GSM and bluetooth devices as global beacons. Skyhook Technologies [17] developed WiFi-based positioning services, which are coupled with smart phones such as iPhones. This essentially takes advantage of the tens of millions of WiFi access points that exist in all major cities, consistently providing moderate location information indoors and in dense urban areas. As a software-only implementation, Skyhooks positioning service does not require additional specialised hardware embedded on the device or installed at the basestation. Google have just released a new service called Latitude [72]. Latitude lets smartphone and laptop users share their location with friends and allows those friends to share their locations in return. Although it cannot pinpoint accurately, Latitude can display your general location based on information from satellites and cell towers. Latitude works on both mobile devices and personal computers.

A wide variety of applications have been developed that utilise location-based services. Without having to disclose their location to others, users can run navigation-oriented applications that display their location on a map, highlight local points of interest, or plot a route to a destination based on current location (e.g. mappoint). Users that are comfortable disclosing their location to their social network have access to applications like dodgeball [9] and mModes' Friend Finder [5] that facilitate social interactions in the physical world. Finally, for users willing to disclose location information to institutions, useful day-to-day services like Googles Local Search and Yahoo Yellow Pages are available to anyone with a network connection.

Sharing location of course largely concerns privacy issues. From a privacy perspective, many of the projects described above have an opt-in service (i.e. you have to ask to get it, it is not provided automatically without asking your permission first) which is good. It also gives a choice of levels of visibility. However, the privacy concerns are similar to that as with the increasing practice of tracking mobile phones today. Apart from the obvious risks to privacy, e.g. everyone getting to know where everyone is, that is if you care, and companies (e.g. Googles' Latitude) holding more information than what they have promised, finally providing yet another vector for surveillance by government authorities.

3.1.2 Requirements, choice of technology and contributions

Clearly these kind of social networking applications, do not need to be extremely precise to the level of *centimetre*, normally room-level or down to few metres accuracy can be acceptable, as largely one needs to know around which area, building or floor the colleague/friend is located. Many friend-finder applications use GSM-based location services offered by the service provider [69]. For example, AT&T Wireless launched a friend finder application before the Cingular acquisition a few years ago, called Find Friends, that service relied on triangulation to locate a subscriber, similarly Garmin devices find buddy beacon locations [69]. The resolution of both are very coarse, although the application does not require precise fine grained location information, it would be beneficial, to get better accuracy when alternative cost-effective solutions are viable, by leveraging the existing infrastructure. Additionally many of these applications may also require information whether the user is moving or still i.e. the mobility status (e.g. if a meeting is scheduled and if the friend is not yet present, knowing his context on whether he is already moving, is likely an indication that he is walking towards the meeting area). The importance of such applications comes partly from the architecture used for sharing the location.

From the requirements listed above and the earlier survey, WLAN technology seems to be the best fit for this kind of application. The desire of using WLAN infrastructure particularly to derive context (i.e. motion and location is very strong, both from the perspective of the availability of the clients device and the infrastructure availability– nearly all the PDA's and laptops have built in wireless interface and there is growing number of mobile phones that are equipped with a WLAN interface as well. Additionally, personal devices like an mp3 player are also equipped with 802.11 radios thus eliminating the cost associated with the tags/locatables and WLAN network infrastructure has become almost ubiquitous with perfect coverage nearly in all the important places of interest. Ekimoto et al. [57] published in 2007 reporting over a half-a million known access points mapped in the Tokyo metropolitan area. These kind of densities offer high coverage with no additional infrastructure.

In Chapter 4 we present algorithms for motion and location inference by leveraging existing WLAN infrastructure. Our contributions include– (i) in-depth characterisation of received signal strength (RSSI), (ii) novel algorithms to deduce motion by observing fluctuations in RSSI across all the access points in range, and (iii) performance comparison using real data against common deterministic location algorithms with and without adding motion information.

3.2 Application II – Inventory and Logistics

The transport and logistic sector plays a major role in the world's economy. As a business concept, it covers the flow and storage of materials from the point of origin to the point of consumption, including inventory management, transport, warehousing and distribution activities. In Europe, the total turnover of the logistics sector in 2006 was estimated to be €800-900 billion [154]. The logistics business is fundamentally all about moving the correct goods from one location to another in the most speedy, reliable and efficient way. It is widely accepted that the real-time location systems (RTLS) drive the penetration of several location-based solutions in transport and logistics–including GPS, GSM, bar code for identification, and other emerging technologies like WLAN, UWB. Several applications such as tracking high value inventory items and personnel in warehouses, ports or manufacturing plants require a precise location information and often in 3D, while certain other applications such as container management or yard management requires accuracy in the order of few metres.

The overwhelming majority of currently deployed RTLS systems use 433 MHz (e.g.

RFCode) or WiFi technology (e.g. Ekahau, AeroScout), with location accuracy typically expressed in metres. The UWB market is very young and small, with few commercial products. Cost and calibration, especially when compared to other alternative solutions are major factors preventing UWB to take a lead. Despite these obstacles, UWB RTLS is seen as a viable, growing technology, primarily because it provides highly accurate location data and works effectively indoors and outdoors. Leading commercial vendors offering UWB-based RTLS are Ubisense [182] and TimeDomain corporation [19] a technology provider manufacturing UWB chipsets.

Table 3.1: *RTLS technology types (Asia Pacific) 2006 (adaped from: [22])*

RTLS Type	**Wi-Fi**	**UWB**	**Passive**	**IR**	**Active**
Cost	$$-$$$	$$$-$$$$	$	$$$	$$$
Power Requirement	High	High	Low: Magnetic Induction	High	High
Battery lifespan(years)*	3-5	3-5	Not required	4-7	1-2
Range	Indoor 60-100m Outdoors 100-200m	Up to 75 m	10-15 m in and outdoors	15 m convergence with RF up to 250 m	10-100m both in and outdoors
Accuracy	1-2 m additional access points likely required. 5-10 m typically	Can be as low as in inches typically 30 cm	5-10 m typically	7-12 m typically	5-10m typically
Continous monitoring	Yes	Yes	No	Yes	Yes

Table 3.1 shows the technology overview for the Asia Pacific market in 2006 [22]. With regard to the percentage of revenues by product type, the WiFi RTLS market is expected to contribute the largest portion of the RTLS market (during the forecast study period) as enterprises are expected to be able to leverage on the usage of WiFi networks, which enables them to also use other applications simultaneously. A report “WiFi finds itself in the real-time location systems market” issued by Research and Markets in April 2006, projected the subset of WiFi RTLS tags dropping from $60 unit price in 2006 but rapidly growing in number (two million tags in 2010) [79]. The UWB and passive markets are growing steadily as these types of RTLS offer distinct advantages. UWB RTLS is expected to be adopted in areas where high level of resolution is required, such as personnel tracking, whereas passive RTLS is expected to be widely used by enterprises that require short to medium range tracking. Passive RTLS

*Highly variable

is also expected to be adopted due its significantly lower overall cost compared to other RTLS types. The IR RTLS market is also expected to grow annually, but growth is not expected to be as significant as other RTLS technologies as this type of RTLS operates only best in indoor environments.

3.2.1 Requirements, choice of technology and contributions

The key features required by the supply chain applications especially for tracking high value inventory items are accuracy (fine-grained granularity, typically less than a metre) and real-time tracking capability which essentially corresponds to systems supporting higher update rates. From scalability point of view, the systems must have the ability to track multiple tags at a faster rate. In most cases the deployment area is a harsh environment and hence the technology must be robust to provide fine-grained location data, despite the deployment area. Additionally, other requirements are: ease of system installation, impact to existing network infrastructure, leverage existing infrastructure, battery life of tags. From the list of RTLS solutions presented in Table 3.1 and the survey presented in Chapter 2, ultra-wideband-based positioning technology could be a possible fit. While calibration cost can be high, the reported accuracy from UWB vendors like Ubisense are down to 30 cm. This is mainly because of UWB's inherent ability to cope well with multipath (in comparison to conventional RF (as explained in Chapter 2)) and theoretically it can give high time resolution to achieve precise ranging.

Chapter 5 addresses this topic in detail. Our contributions include–(i) characterisation of heterogeneous observations (pseudoranges and angles) obtained from two deployments of Ubisense (a commercial UWB positioning system), mimicking real-world vs. ideal deployment, (ii) formulation of algorithms to fuse heterogenous observations (iii) a thorough evaluation for both static and dynamic tracking (iv) showing the effectiveness of the algorithm when applied to work on homogeneous data.

3.3 Application III –Emergency Response

Search and rescue is a challenging and dangerous activity. The environment is often unfamiliar, changing and visibility can be limited. The rescue operations are time-critical and hence quick decision making support and close coordination within rescue teams are required. Ad hoc tracking and navigation support for emergency response is an important and safety-critical challenge. A report on the Worcester warehouse fire, in which six firefighters died, highlights the difficulty to keep track of firefighters within the building as one of the major causes for loss of lives [30]. Fahy report [59] on fatalities in structure fires linked 29 casualties between 1990–2000 to firefighters becoming lost inside the structure. The application pull for new technologies to address safety of emergency responders is evident in major initiatives including fire services, fire protection agencies and relevant industries [129, 177, 200] but new research is required to tackle the problem of ad hoc tracking and navigation. Fire accidents may cause severe damage to the existing sensing, positioning and communication infrastructure. Many casualities are caused because of lack of communication between the firefighters and the incident commander [98]. This calls for the need for having an ad hoc wireless network, which can be *quickly set up* at the disaster site to provide a reliable communication link between the firefighters inside and the incident commander.

There are many new technologies that can be used for enabling better communications for first responders. Researchers at UCSD have created lightweight nodes that are called *WiFi bubbles* to communicate data from the disaster scene [23, 161]. There are also other similar works using different technologies to create *disposable nodes* for establishing communication capabilities at the disaster site [98]. If the same ad hoc network can provide sensing capabilities, it could be used to relay vital data from the firefighters wirelessly. For instance if the firefighter has stopped moving, help can be called in immediately [23, 143]. The same network could also collect environmental information such as temperature, air quality and visibility information on the way back using this information the firemen can be warned if the way they entered is blocked. Lastly, knowing the location together with the status of the firefighters helps in efficiently managing the deployment teams and navigational support allows the firefighters find their way back when entering an unknown building with impaired visibility [103]. Therefore, an ad hoc wireless sensor network is crucial for improving the effectiveness of the rescue operation and for saving lives. The deployed sensor network can be used for providing communication and environmental sensing in addition to providing positioning capabilities. Our work focuses on providing tracking and navigational support for the responders, and will be explained in more detail in Chapter 6.

3.3.1 Lifeline navigation

When entering a building with poor visibility because of fire, firefighters use ropes to mark the path taken. These ropes are called *lifelines* (see Figure 3.2) help them to most importantly find their way back and help other teams to take the same path. Furthermore, pulling the line can signal people outside the fire area. While this technique works reliably in many situations, there are also a number of shortcomings that can become critical and in fact have led to fatalities in the past. In particular, the lifeline can get stuck or be cut under doors or other objects. It can also become entangled with furniture, railings etc. and generally it limits the operational range. Moreover, a lifeline always offers only one retreat path and communicates none or very little information about the firefighters to the outside. Klann [102] details on the design of a wearable system that can provide navigational support to firefighters.

3.3.2 Requirements, choice of technology and contributions

Tracking the firefighter inside a burning building is critical for incident commanders to assess the situation, to allocate more personnel assisting and aiding the rescue operation, and to provide guidance support for the responders to find their way back to safety in low visibility. Guidance may also be needed to direct responders towards victims that have already been located by other teams. Localisation in such dynamic environment is difficult because no infrastructure can be presumed, as it might have been partly collapsed or completely unusable due to the advent of fire. The ideal solution for immediate intervention is to have an auto-deployable system that operates as a standalone system providing reliable positioning and communication infrastructure. The disaster that happened at the Worchester warehouse shows that some firefighters have lost their life, while they were just few feet from the exit [30]. So, such systems require high and fine-grained location accuracy (less than a few metres). Second major issue pertaining to the development of such system, is *the size of the device* to be deployed and the ease with which they could be deployed, as firefighters would

Figure 3.2: *Paris firebrigade training.*

already be carrying around 18 to 28 kg equipment [95] and the deployment should not be too time-consuming.

Technology	Advantages	Disadvantages	Example
Ultra-wideband (UWB)	Accuracy	Deployment and calibration	ThalesIPS [73]
Radio frequency Identification (RFID)	Robust	Tags or readers to be predeployed, map layout	EPFL [166]
Inertial sensing systems (INS)	Autonomous positioning system	Large drift errors, communication reach back	Navshoe [67]

Table 3.2: *UWB, RFID, INS technologies — pros and cons.*

Beauregard et al. [34] gives the following requirements for the location system in a worst case scenario: accuracy of 1 *to* 2 *m* or room level, update rate faster than 1 *Hz*, range from last known reference point of 100 *to* 500 *m*. Field studies in the Siren project [96] and the FIRE project [177] have shown that fine-grained location is not a priority and that *reliability* is more important for firefighters. Table 3.2 lists some of the possible list of technologies for providing solution to the first responders. Applying our taxonomy to make choices for this application, clearly the location accuracy should be fine-grained, in order to be able to distinguish which side of the wall the responder is located. The choice of architecture or deciding where the computation should occur is not crucial. The system must provide reasonable update rates. Cost (in terms of tag/beacon) and privacy are clearly not a concern when it comes to a matter of saving lives. However, the chosen system must not have high calibration cost. From the requirements and based on the state-of-the-art systems, clearly we can rule out many location systems like GSM, GPS or WLAN-based. UWB systems on the other hand provide sufficient accuracy for the intended application, but most of the state of the art UWB systems do require an extensive amount of calibration before putting the system to use, hence we could rule that out. Inertial sensors provide completely autonomous capability and hence seems to be the best fit for this specific application. However, since inertial sensors are inherently affected by drift, alternative external sources are to be used for correcting the drift. Looking for external sources, ultrasound sensors seems to be a promising source. Ultrasound-based positioning systems have been reported to provide fine-grained position accuracy of upto 10 cm and orientation accuracy of 30° [82]. The US beacons provide more accurate distance measurements than the dead reckoning and can correct for drift. They also provide a multihop communication network which virtually follows the responders and flashing lights and audible signals on the beacons can provide a fallback guidance solution [40]. However, there is always a problem of blind spots or dead zones where one technology seize to function. Ultrasound signals could get blocked in certain parts of the building, or the device might lose its functionality. For instance, if a firefighter is either out of range with a US beacon or if the beacon in proximity is not functional, inertial navigation could then be used as a stand alone system providing positioning and tracking functionality. In combination, the two technologies would add inherent robustness to the system. Such a combination will also result in improved update rates, as inertial sensors have significantly higher update rate (typically over 100 Hz [199]) than the narrow band ultrasonic sensors.

We detail the usage of ultrasound together with inertial sensors for providing tracking and navigation support to firefighters in Chapter 6. Our contributions include, (i) characterisation of inertial and ultrasound data and (ii) algorithms to support tracking and guidance (iii) thorough evaluation from data gathered from real deployments.

Application	Dep.	Coverage	Range	Tech.	Meas.	Arch	Comp.	Output rep	Obj. Ass.	**Estimation**
Social networks	Inf	In/Out	Long	WLAN	RSSI	Passive	Loc.	Phy/Sym	Tagged	**Chapter 4**
Inventory & Logistics	Inf	In/Out	Med /Long	UWB	TDOA +AOA	Active	Cent	Phy/Sym	Tagged	**Chapter 5**
Emergency Response	Ad hoc	In	Short /Med	US +inertial	TOA +AOA +inertial	Active/ Passive	Cent	Relative	Tagged	**Chapter 6**

Table 3.3: *Requirements for three chosen applications derived from our taxonomy (presented in Chapter 2). The column "Estimation" is detailed in the corresponding Chapters.*

3.4 Conclusions

The progress and development in location systems have always been triggered by the needs of the application. We have outlined three specific applications – (i) location and tracking for social networking, (ii) inventory and logistics in warehouses and (iii) emergency response –

that are of direct relevance to the work presented in this thesis. From the earlier presented taxonomy we were able to create a blueprint of a location system that would meet those three application needs. We have outlined at the end of each application, our specific contributions.

Table 3.3 and Table 3.4 summarise based on the earlier defined taxonomy the requirements of all the three applications. The last column in Table 3.3 corresponds to the core of this thesis – localisation algorithms. While the chosen technologies and applications are not exhaustive, they are representative as they cover a broad spectrum across several dimensions: *accuracy* – fine-grained to coarse-grained, *coverage* – room-level to wide-area, *dependence* – dense infrastructure to ad hoc, *cost* – expensive to minimal cost, and in every instance, we illustrate the benefit of combining multiple modalities.

Application	**Required Performance Measure**								
	Accuracy	Update	Latency	Privacy	Cost			Scalability	
					Inf-cost	Tag-cost	Calib. cost	Inf-scale	Tag-scale
Social networks	coarse	medium	medium	yes	n/a	n/a	low	high	high
Inventory & Logistics	fine	high	low	no, opt-out	high	medium	high	high	high
Emergency Response	fine	high	low	does not matter	n/a	n/a	low	n/a	n/a

Table 3.4: *Performance measure required by the application (as defined by taxonomy Section 2.2).*

CHAPTER IV *

Inferring motion and location using WLAN RSSI

As we outlined in Chapter 3, applications such as social-networking do not require precise fine-grained location information. Such applications weigh more on solutions that can offer *cost-effectiveness* and *larger coverage*. With this as the motivation, in this chapter we present *algorithms* that can infer *motion* and *location* by leveraging the existing WLAN infrastructure. We first outline the contributions of this chapter, then focus on characterising signal strength, which forms the basis for the algorithms presented in this chapter. After a thorough evaluation of the motion detection algorithms against traces of data gathered over a longer period of time from diverse environments, we present the effect of combining this motion detection scheme as part of WLAN location algorithm to see the benefits of adding this extra context. In this pursuit, we compare two existing location algorithms, the Centroid and Weighted Centroid algorithm, as opposed to popularly used *radio fingerprinting* based methods. We show the effectiveness of adding the motion information obtained from our motion detection algorithm to smooth the stream of location estimates and hence show improvements in accuracy. In addition, we present a privacy observant architecture to share location information, where we handle the location of people as services which can be easily discovered and used.

*This chapter combines the following three publications: "Inferring motion and location using WLAN RSSI" In the Proceedings of Mobile Entity Localization and Tracking in GPS-less environments (MELT), Orlando, USA, September 2009 [141], "Sensing motion using spectral and spatial analysis of WLAN RSSI" In the Proceedings of European Conference on Smart Sensing and Context (EuroSSC), Lake District, UK, October 2007 [139] and "WLAN location sharing using privacy observant architecture" In the Proceedings of Communication System Software and Middleware (COMSWARE), New Delhi, India, January 2006 [140].

4.1 Introduction

Ubiquitous computing is emerging as an exciting new paradigm with a goal to provide services anytime anywhere. Context is a critical parameter of ubiquitous computing. Ubiquitous computing applications make use of several technologies to infer different types of user context. The context cue that we are interested in is users' *motion* being either *"moving"* or *"still"* and *location*.

There are plethora of ways to derive mobility information ranging from using accelerometers to specialised motion capture suits. While accelerometers and motion sensors were once considered as an additional hardware, now with the advent of smart phones and more sophisticated laptops, accelerometers are provided for protecting system operations and to support the user interface (such as automatic adjustment of screen orientation). With accelerometers it is still possible to generate false readings, for instance by shaking the device when the user is still. Specialised motion tracking and motion detection systems make use of a range of sensing technologies [66]. Although such dedicated systems typically provide highly accurate information concerning position and even orientation they require additional hardware, which is often unwieldy, impractical or simply not available. Theoretically it is possible to compute motion by differentiating measured location over time, however considering the unreliability and uncertainity in the sensor data, deriving motion from location is not always possible. Alternatively, solutions which use existing infrastructure to determine the state of the user are gaining popularity [29, 110, 123, 174].

The desire of using WLAN infrastructure particularly to derive context (i.e. motion and location) is very strong, both from the perspective of the availability of the clients device and the infrastructure availability–nearly all the personal digital assistants (PDAs), laptops have built in wireless interface and there is a growing number of mobile phones that are equipped with WLAN as well. Additionally, personal devices like an mp3 player are also equipped with 802.11 radios. WLAN network infrastructure has become almost ubiquitous with perfect coverage in many important places of interest, Ekimoto et al. [57] reports over half-a million known access points mapped in the Tokyo metropolitan area. These kind of densities offer high coverage with no additional infrastructure. Location-based services, using WLAN tags in settings like hospitals to track equipments and patients are currently the trend [56, 3] and localisation using WLAN is considered as a value added application to WLANs [33, 56].

Looking at the applicability and usefulness of motion detection, the WLAN radio by itself can sense motion, and it can potentially be also part of the sensor ensemble to improve recognition performance. Apart from inferencing activity of the user itself, it has been showcased that such motion inference is useful for efficient fingerprinting solutions [37]. Recently movement detection was shown to adaptively switch between passive sniffing and active scanning to allow positioning and to minimise the impact on the communications [99]. In this work, we show yet another use of incorporating motion detection as part of localisation algorithm. The applications that are described above do not necessarily benefit from accurate and complete information about the mobility status. For the purposes described above it is sufficient to know whether the user is moving or not.

This chapter examines the results of several motion and location sensing algorithms that operate on RSSI data gathered from an existing WLAN infrastructure. The main advantages of the proposed algorithms are: *(i)* deducing users context using the existing infrastructure (*WLAN*) without a need of additional hardware, as it offers a pure software-based solution and *(ii)* preserving user privacy, as context inference can be performed locally at the client device.

Contributions: The key contributions of this chapter are as follows:

- A detailed characterisation of WLAN RSSI measurements, exposing a rich set of features both in time and frequency domain to gather mobility information. Our analysis in both temporal and spectral domain, results in a conclusion that "when a device is moving, signal strengths of all heard access points exhibit higher variation compared to when a device is still and the number of detectable samples from access points varies considerably when the device is moving".
- We present novel algorithms to infer movement that makes use of inherent fluctuations in the signal strength. We evaluate the performance of the presented algorithms thoroughly based on classification metrics such as *recall* and *precision* from annotated traces (typically groundtruth recorded for every second) obtained over twelve hours in total from different types of environment and with different access point densities.
- We show that how common deterministic location algorithm such as Centroid and Weighted centroid can improve its accuracy when a motion model is included. To the best of our knowledge, a motion model is normally used only in probabilistic algorithms and such simple deterministic algorithms have not used a motion model in a principled manner. We evaluate the performance of algorithms against traces of RSSI data collected from different environments, with and without adding mobility information inferred from the mobility detection algorithm.

4.2 Related work on motion sensing

In this section, we briefly outline the relevant work in methods used to sense motion.

Randell et al. [162] demonstrated the possibility of distinguishing various states of the movement such as walking, climbing and running using a 2D accelerometer. Patterson et al. [123] take the velocity readings from GPS measurements and infer the transportation mode of the user, for instance walking, driving, or taking a bus using a learning model. The model learns the traveler's current mode of transportation as well as his most likely route, in an unsupervised manner. It is implemented using particle filters and is learnt using Expectation-Maximisation. The learnt model can predict mode transitions, such as boarding a bus at one location and disembarking at another.

A promising alternative to usage of specialised hardware is to investigate what can be obtained by measuring signals received from existing infrastructures (either WLAN or GSM). Since WiFi access points and WiFi clients are ubiquitous, this technique is very attractive. In this line, Krumm et al. [110] classified a user as either moving or still based on the variance of a temporally short history of signal strength from currently the strongest access point. This classification had many transitions, hence it was smoothened over the time/period with a two-state hidden markov model (HMM) resulting in an overall accuracy of 87%. Privacy is enhanced compared to systems that compute context on a central server, since the context inferences rely only on client-side data and computations.

Anderson et al. [29] use GSM cellular signal strength levels and neighboring cell information to distinguish movement status. The classification of the signal patterns is performed using a neural network model, resulting in an average classification accuracy of 80%. The authors trained the neural network initially and demonstrated a proof of concept by executing it in real-time on a cell phone. However, the initial training did not work in all the environments as signal strength fluctuations were different.

Recently Sohn et al. [174] published a similar technique for detecting user's motion using signal traces from a GSM network. Their motion detection system yields an overall accuracy of 85%. They extracted a set of seven features to classify the user state as either still, walking, or driving.

Our work on motion detection algorithm is similar to the work on Anderson et al. [29] and Sohn et al. [174], but we look into variation in the WLAN RSSI observed across several access points as opposed to GSM signals.

4.3 Preliminaries

A premise of our work is that signal strength information provides means of inferring device location and motion status. In this section we briefly outline the typical WLAN network configurations and scanning methods used for gathering RSSI measurements.

WLAN Primer

WLAN can work in two different types of network configuration – *ad hoc* and *infrastructure*. Ad hoc networks are created by two or more wireless enabled devices communicating directly with each other and are useful in creating small, dynamic networks. When operating in infrastructure mode, the network consists of both the *infrastructure entities*, i.e. access points and *mobile entities*, i.e. clients devices. The 802.11 protocol includes beacon frames that are periodically* sent to the client device by the access point, indicating its presence. The device can then passively learn the information from the access point (i.e. passive scanning/sniffing) or alternatively, the device can initiate an active scan by sending a probe request to the access point and waiting for the probe response frames that are sent back (i.e. active scanning).

Sampling RSSI

The IEEE 802.11 standard defines a mechanism by which RF energy is to be measured by the circuitry on a wireless NIC. In 802.11b/g/a, this numeric value is an integer with an allowable range of $0 - 255$, called the RSSI. 802.11 does not require that a chipset vendor use all 255 values, so each vendor will have a specific maximum value. For example, Cisco chooses RSSI-max as 100 while the Atheros chipset uses 60 as the maximum value. These values do not correspond to RSSI in dBm – these values are used internally by microcode on the WLAN card and by device drivers to report the quality of the signal. The mapping between the RF energy levels and the range of RSSI values can be done, but is vendor specific. We use the

*The period/frequency of the frames differ, but is typically about 102 ms.

RSSI values reported in dBm by manufacturers for the analysis presented in this chapter. All the WLAN-based location systems make use of 802.11 radios and their supporting drivers allow the device to scan for the nearby access points. Thus, a typical scan would return the MAC address of the access point, the received signal strength and the SSID (as shown in the Table 4.1) and is called a *spotter*†. We use an openNetCF.Net.dll library [24] for gathering the signal strength and a sampling rate of 0.4 Hz (maximum sampling rate that is achievable on a device operating Windows Mobile 5.0/6.0) is used.

AP BSSID	SignalStrength	SSID
000b5fd00de8	-75	WLAN
000b5fbcc0e0	-91	WLAN
000b5fd7f214	-88	WLAN
000b5fd00d2e	-82	WLAN
000b5fd7f1c5	-45	WLAN
000b5fd7f1d6	-61	WLAN

Table 4.1: *An example of a scanning result.*

4.4 Characterisation of received signal strength (RSSI)

In this section we investigate some of the properties showcased by RSSI, particularly when the device is “still” and “moving”.

Temporal characterisation

This subsection describes how the RSSI changes over time when the user is still and moving, both in static and dynamic environments. By static environment, we mean when the device is placed in a relatively stable environment (e.g. by logging measurements at off-peak hours) and dynamic environment refers to an area which is affected by moving people (e.g. a canteen during lunch hours). Figure 4.1 shows an example of temporal characteristics. We can observe from the figure that only some of the access points show a clear distinction between the “still” and “moving” periods– specifically the weaker signal strengths (RSSI <-75dBm) do not convey any significant difference for both still and moving, hence we have used only the stronger access points for the analysis presented here.

The main intention of logging a“still-moving-still” phase taken in a static environment and “moving-still” phase taken in a dynamic environment is to see if the variations in the signal strength is influenced more by the changing environment around a static device or by the device movement. We do observe larger stability in the RSSI values for measurements recorded both at static and dynamic environments when the device is still, in comparison to the moving device. Also it is noticeable in Figure 4.1 that when the user is still, the dip in the signal‡ occurs as bursts lasting for a very short duration. However, for the moving case the variations in the signal occur more persistently. But in some cases, we do observe erratic

†Placelab terminology [12]

‡Throughout the text presented in this chapter, we use the terminology RSSI, signal strength and signal interchangeably.

behavior in RSSI measurements even when the device is still, and hence it is rather difficult to generalise these conclusions.

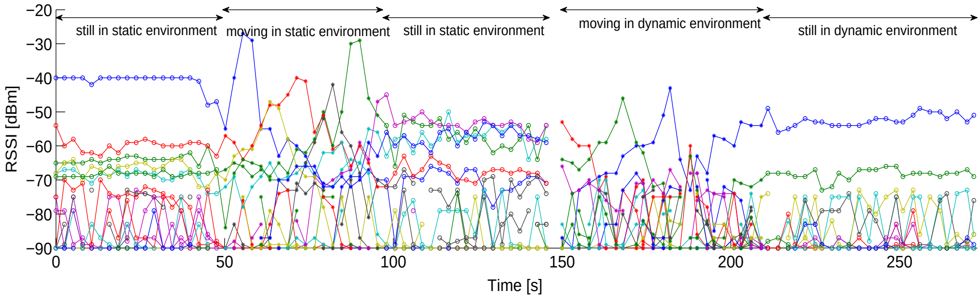


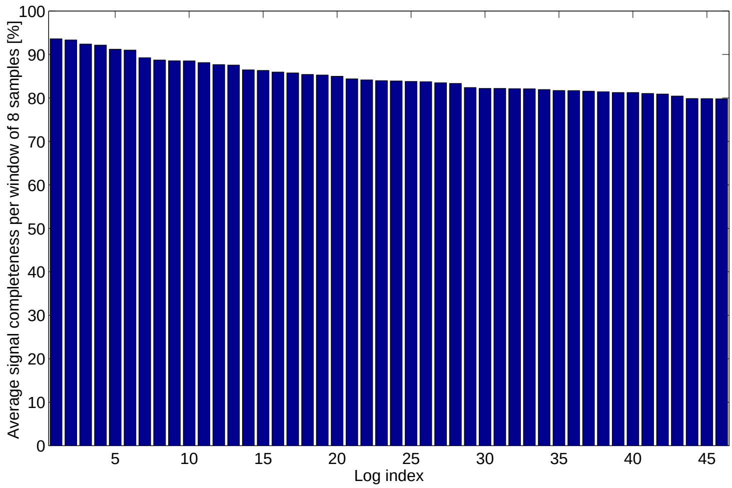
Figure 4.1: *This figure illustrates a "still-moving-still" phase that is measured in a static environment and "moving-still" phase measured in a dynamic environment over a period of four minutes. Each of the lines represent signal strength received from a specific access point.*

But more importantly, we observe a lot of variations in the number of samples received within a particular observational window (e.g. window of eight samples) as shown in Figure 4.2. It is particularly interesting to note that, at a fixed location the number of signal strength samples received from the same access point over a window of reading fairly remains closer to 85% on an average. This is reasonable, as in one scan we typically do not hear all the access points, so one or two missing signal values is still relatively acceptable when the device is still. As opposed to this, in the case of moving, the number of signal strength samples received from the access point varies as the number of access points detectable at a place varies greatly as the user moves. Each of the bins in Figure 4.2 represents the average result of the number of samples seen from all the detected access points over all windows of eight samples from one distinct log. In total for still and moving, we collected over ninety different logs with average duration of log spanning for about seven to eight minutes.

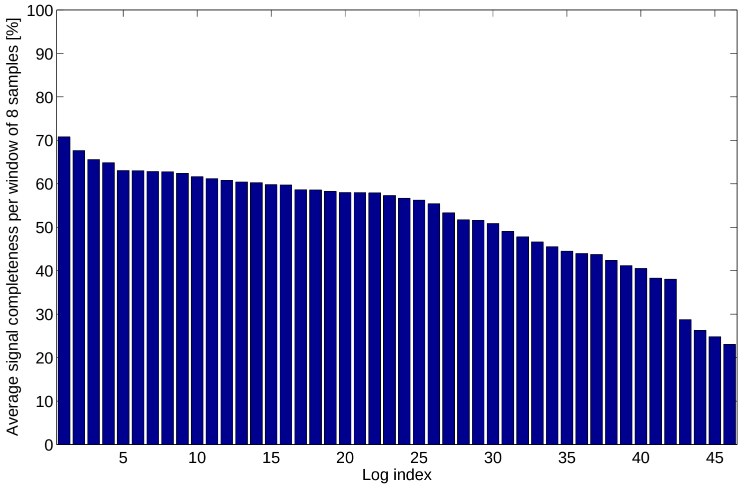
Spectral characterisation

In this subsection we look at the spectral characteristics of RSSI measurements. The schematic representation of how a signal looks in time and frequency domain is illustrated in Figure 4.3(A). As a rule of thumb, the more concentrated the time domain, the more spread out the frequency domain. In particular, if we "squeeze" a function in time, it spreads out in frequency and vice-versa. Also Figure 4.3(B) illustrates full width at half maximum (FWHM) that corresponds to peak width of the FFT signal at 50% peak height.

To view the frequency representation, we apply Fast Fourier Transform (FFT) to the



(a) Still



(b) Moving

Figure 4.2: *Variations in the number of samples received when (a) Still and (b) Moving. Each log (with an average duration of seven to eight minutes) is split into window of eight samples and the results are averaged together.*

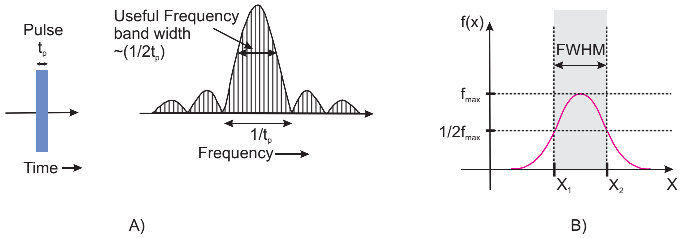


Figure 4.3: *(A). Schematic representation of a rectangular pulse in time and frequency domain, short duration pulses produces a large bandwidth (B). Full Width Half Maximum, corresponding to peak width at 50% peak height. t_p is the pulse period, and f_{max} is the peak at maximum. X_1 and X_2 are used for calculating FWHM (explained later in Section 4.5.2).*

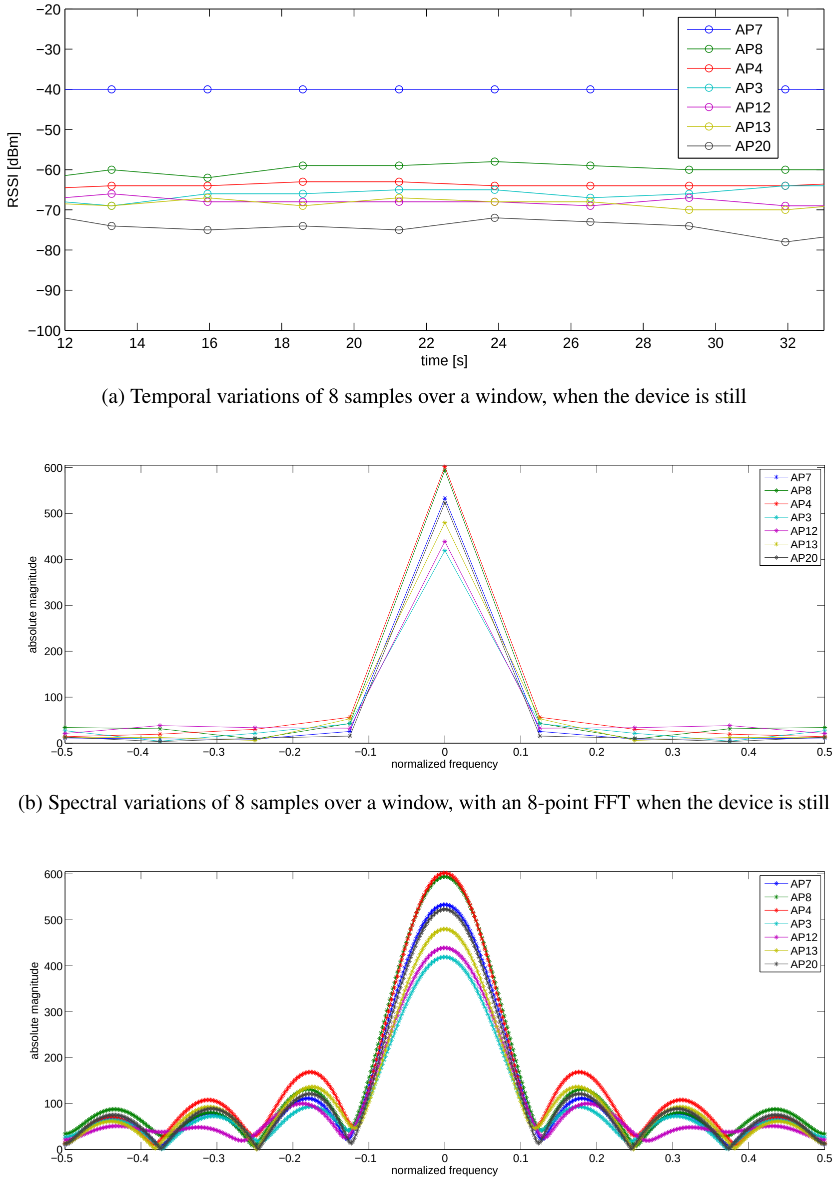
WLAN RSSI in time series. The FFT of N points x_n is defined as follows:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-2\pi ikn/N} (k = 0, ...N - 1) \tag{4.1}$$

where X_k is the k^{th} coefficient of the FFT and x_n denotes the n^{th} sample of the time series which consists of N samples and $i = \sqrt{-1}$. The typical syntax for computing the FFT of a signal, $FFT(x_n, N)$, where x_n is the input signal we wish to transform (RSSI in our case) and N is the number of points in the FFT. N must be at least as large as the number of samples in x_n.

Figure 4.4 (a) and Figure 4.5 (a) presents temporal variations in the signal strength observed over a short window of eight samples (approximately twenty seconds duration) from the strongest seven heard access points when the device is still and moving. The corresponding frequency domain representation is shown in Figures 4.4 (c) and 4.5 (c). It is evident that although signal strength varies even while the user is still, this variation is reflected in all the heard access points uniformly as there is a well defined peak with a narrow spectral width in the frequency domain from all the access points. Although there are differences in the Fourier amplitude from each of the heard access point, the spectral width seems to be broader for moving cases. But when the user is moving, there is no well defined peak from all the access points in the frequency domain indicating that variation in the signal strength happens more often, and not in all the heard access points in the same manner. Specifically, we observe the effect of spectral broadening from a significant number of access points when the user is moving, resulting in a wider full width at half maximum. This phenomenon happens mainly due to two reasons: *(i)* the variation in the signal strength is large in case of a moving user and *(ii)* the number of access points detectable varies with distance resulting in too few received samples from the access points. This confirms that both the temporal and spectral analysis lead to the similar conclusions but give a different view of representation.

To demonstrate the effect of changing value of N, let us compare Figures 4.4 (b) and 4.5 (b) (where N = 8) with Figures 4.4 (c) and 4.5 (c) (where N = 512). One can see that in

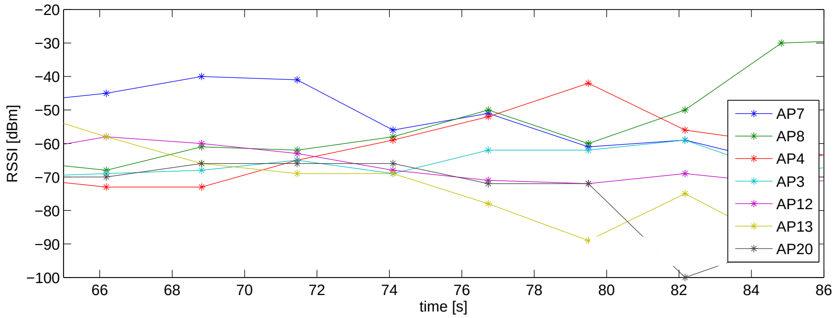


(a) Temporal variations of 8 samples over a window, when the device is still

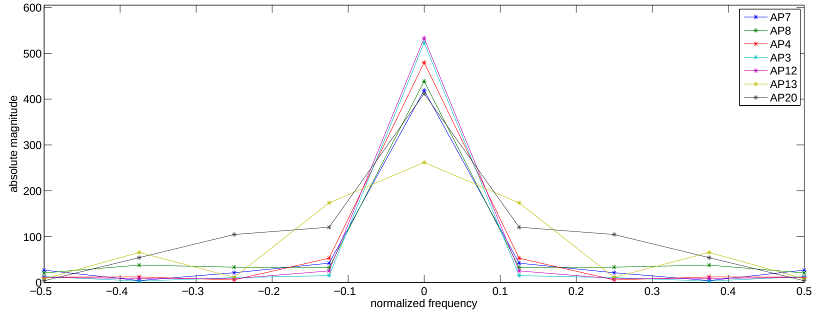
(b) Spectral variations of 8 samples over a window, with an 8-point FFT when the device is still

(c) Spectral variations of 8 samples over a window, with a 512-point FFT when the device is still

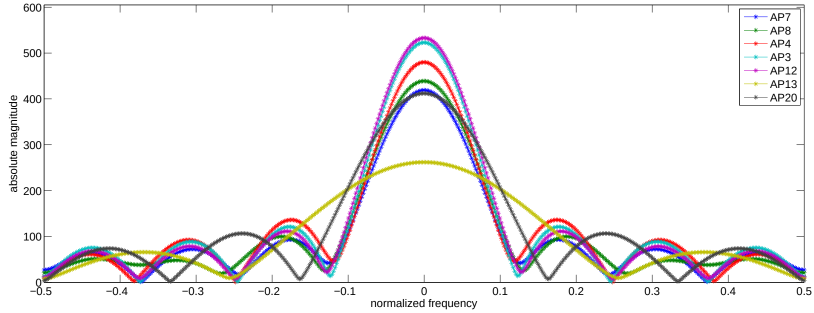
Figure 4.4: *Temporal and Spectral characteristics of a window of eight samples of the strongest seven access points for the case of "still". (a) The signal taken is a subset of the signals that are represented in still phase in Figure 4.1 for time varying between 12 – 32 s, and corresponding FFTs in (b) and (c). For (b) and (c), the frequency scale is normalised and shifted to extend it from* -0.5 *to* $+0.5$.



(a) Temporal variations of 8 samples over a window, when the device is moving



(b) Spectral variations of 8 samples over a window, with an 8-point FFT when the device is moving



(c) Spectral variations of 8 samples over a window, with a 512-point FFT when the device is moving

Figure 4.5: *Temporal and Spectral characteristics of a window of eight samples of the strongest seven access points for the case of "moving". (a) The signal taken is a subset of the signals that are represented in moving phase in Figure 4.1 for time varying between 66–86 s, and their corresponding FFTs in (b) and (c). For (b) and (c), the frequency scale is normalised and shifted to extend it from* −0.5 *to* +0.5.

each case, the transform adheres to the same shape differing only in the number of samples used to approximate that shape. Note that for $N = 512$, the number of actual signal samples used is only 8, but the rest of the 504 samples are padded with zeros. For both the cases ($N = 8$ and $N = 512$) the frequency scale begins at 0 and extends to $(N - 1)$ for an N-point FFT. Because we are only interested in the frequency amplitudes and not phases, we take the absolute value of the complex FFT-values. We then normalise and shift the scale so that it extends from -0.5 to $+0.5$, such that the FFT is symmetric around zero.

4.5 Algorithms for sensing motion

In this section we present algorithms for sensing motion. The algorithms are categorised into time domain and frequency domain based on the observations presented in Section 4.4. All the algorithms that we explain here are based on "thresholding" applied to a certain metric. The details of how these thresholds are obtained is elaborated in Section 4.6.2.

4.5.1 Time domain algorithms

We use four different types of metric that we observe in the temporal domain to infer user movement. As opposed to looking at one RSSI value, all the algorithms presented here use RSSI observed over a window of readings (window size = eight samples). It is important to highlight how different metrics used to distinguish user states can vary depending on the measurement environment (indoors/outdoors). For this reason, we have represented (Figure 4.6 – Figure 4.8), traces of data captured outdoors (ranging from 0 – 300 s) and indoors (ranging from 300 – 600 s).

AP Visibility This is the simplest algorithm as it just uses the proportion of the time that RSSI of a particular access point is observed within the observation window. Figure 4.2 already gave an indication of its potential. The proportion of time for which each access point is observed is calculated and then averaged together. Depending on a certain threshold the algorithm detects the state as either moving or still. These thresholds are determined based on the training dataset as explained later in Section 4.6.2.

Rank correlation coefficient We estimate the rank correlation coefficient using Spearman's Rank Correlation Coefficient (ρ) [191]. The rank correlation coefficient between any two measurements represents how closely the signals are ranked. It is defined as the following:

$$\rho = 1 - \frac{6 \times \sum d_i^2}{n(n^2 - 1)} \tag{4.2}$$

where d_i is the difference in rank of the signal measurements and n represents the number of measurements in the data set. ρ takes the value between -1 and 1. Ranking closer to 1 indicates that the measurements are similar and hence the user is still and when the user is moving the ranking is lower. Figure 4.6 presents how the rank correlation coefficient varies when the device is still and moving. This algorithm tracks Spearman's rank correlation coefficient between the first and the last measurement in a observation window as a metric to distinguish between moving and still states.

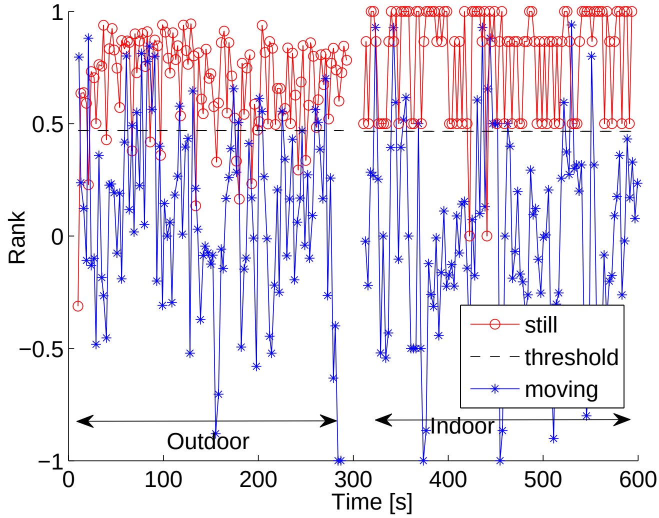


Figure 4.6: *Spearman's rank correlation coefficient when still and moving, for outdoor (left) and indoor (right) environments. The difference in the rank correlation coefficient remains the same for both outdoor and indoor.*

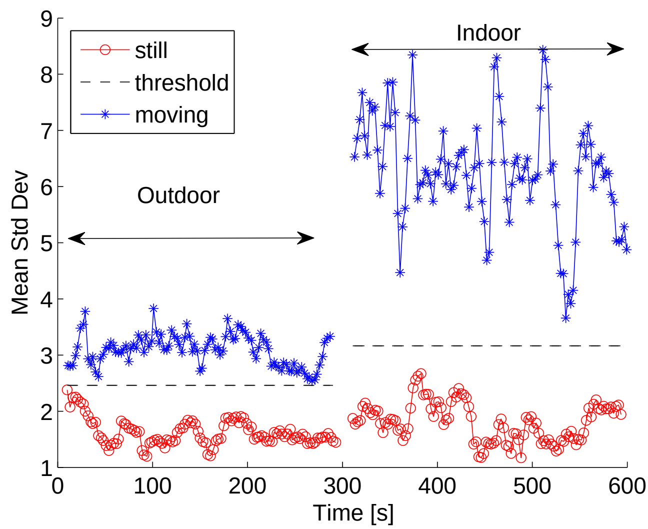


Figure 4.7: *Mean standard deviation when still and moving, for outdoor (left) and indoor (right) environments. Here we can observe a considerable difference in the Std Dev values between the measurements logged from an outdoor and indoor environment.*

Standard deviation This algorithm uses mean standard deviation over all the heard access points as a metric to distinguish between still and moving states. That is, within the observation window we measure the standard deviation (Std Dev) between the measurements for each detected access point, and use the average Std Dev over all heard access points for inferring the motion status. Figure 4.7 presents how the average Std Dev varies when the device is still and moving.

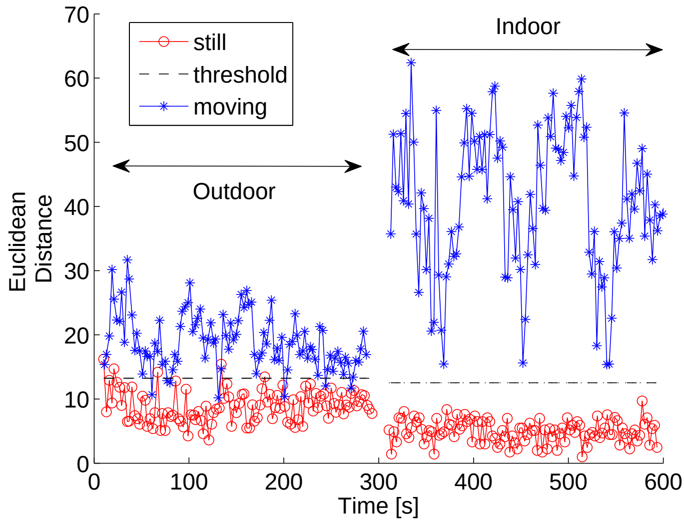


Figure 4.8: *Euclidean distance when still and moving, for outdoor (left) and indoor (right) environments. Here we can observe a considerable difference in the Euc dist values between the measurements logged from an outdoor and indoor environment.*

Euclidean distance We are interested in seeing signal strength fluctuations in Euclidean space, in particular to see whether any signal correlation exists for spatially separated signals for the case when the device is moving and when the device is still. The observation is based on the same principle as fingerprinting-based location systems [33], which state that the signals observed from the access points are consistent in time but variable in space. This has also been used as a means to distinguish motion state by using GSM radio signals [174]. Figure 4.8 illustrates the average Euclidean distance between WLAN measurements and shows that the average Euclidean distance between WLAN measurements are proportional to the state of the movement. When the user is still, the Euclidean distance is relatively small (< 10), when the device is moving the Euclidean distance is higher (typically > 10). In a generalised form, the Euclidean distance between any two measurements can be written as:

$$\sqrt{\sum_{i=1}^{n}(S_{i(X)} - S_{i(X+C)})^2} \tag{4.3}$$

where i is the index of the access point used, X is the index of measurement and C is the measurement interval, $C = 1$ represents consecutive measurements.

The algorithm tracks the Euclidean distance between the WLAN readings using a window of WLAN readings, as defined by the window size. This is inspired by the work presented by Sohn [174] for classifying motion status based on GSM data traces. Sohn et al. used seven different features and machine learning algorithms to train and test for classifying still, walking and driving. Since we are interested in using only a two state classification (i.e. to distinguish between still and moving) we use Euclidean distance over a window of measurements, where the values are calculated between the first and last measurements within the observation window.

4.5.2 Frequency domain algorithms

In this subsection we present novel motion detection algorithms which are based on our observations presented in Section 4.4.

FWHM Thresholding The input to the algorithm is the WLAN signal trace and the corresponding threshold. We filtered out the signals that are weaker than $-75dBm$. The RSSI in time series is converted into the frequency domain using Fast Fourier Transformation (FFT). For calculating the full width at half maximum i.e. peak width of the FFT signal at 50% peak height for each of the access point entries (excluding the filtered access points), we first normalise the signal input and then find the centre index or the peak (refering back to Figure 4.3, f_{max} is the peak of the FFT of the input signal). Following the signal downward from the peak, we find X_1 and X_2 at half the maximum amplitude. Typically, there is no value at exactly this amplitude, hence we interpolate between the two points nearest to it on either side. The difference in X_1 and X_2 essentially corresponds to the *width* at half maximum. For classification, this algorithm uses the FWHM of the main peak of the FFT for a given window of samples and median over all the access points in that window.

FWHM Count This algorithm is very similar to the previous algorithm. The algorithm tracks how many access points have a spectral width that is exceeding a certain threshold within the window of readings. From our earlier observations it is intuitive that a larger spread indicates that the device is moving. The input to the algorithm is the WLAN signal trace and the threshold for FWHM and a proportion of access points that exceeds FWHM threshold from the list of access points observed.

The RSSI is transformed using FFT to frequency domain and FWHM is calculated as explained above. To make a decision based on the full width at half maximum, a threshold is set. Whenever an entry (access point) exceeds the FWHM threshold, the algorithm treats this as an outlier and increments a counter. If the counter exceeds a certain threshold relative to the total number of observed access points, the algorithm returns the user state as *moving*, otherwise it returns *still*. The counter indicates how many of the access points that are detected have a spectral width exceeding the FWHM threshold.

Low-amplitude-frequency count A signal that is not varying much in the time domain has a frequency spectrum with a narrow peak around zero and very low amplitudes at higher frequencies. In contrast, a signal that significantly varies in the time domain has a broader frequency spectrum, i.e. the peak around zero is wider and amplitudes at higher frequencies are not as low as for less varying signals. This can also be observed in Figures 4.4 and 4.5.

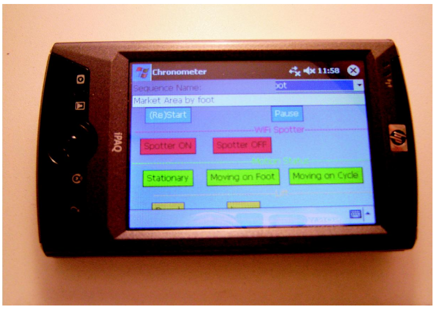
Based on this observation we use a metric, (*low-amplitude-frequency count* or *LAFC*) that distinguishes between "still" and "moving". The algorithm *LAFC* operates on the fast Fourier transformed signal and effectively counts the number of frequencies that have low amplitude (based on earlier experiments we define " low amplitude" as less than 10% of the maximum amplitude in the FFT, as this gave the best results). If this number exceeds a certain threshold, the motion status is set as "moving", and otherwise it is set as "still". The LAFC is determined for each heard access point within the observation window and then averaged over all heard access points.

4.6 Performance evaluation

In this section we present the data collection process and report the performance of the presented motion detection algorithms.

4.6.1 Data collection

Two members of our research team collected WLAN network traces. Each data collector carried an HP iPAQ pocket PC running spotter (RSSI scanning software) for recording readings from nearby access points and logging them. Data collectors recorded their mobility activities using a custom diary application running on the PDA that allowed them to indicate whether they were walking, driving, cycling or staying still. Each time a measurement of spotter was logged, the associated activity performed at that instance was recorded as groundtruth manually (refer Figure 4.9) from the pull down menu of the logging application to make the diary logging more accurate.



Measurement Fri May 25 12:14:44 CEST 2007

25-05-2007 12:14:45 149 Start Sequence:Waaier
25-05-2007 12:14:45 149 Motion:Moving
000136079de0,-90
000cf6164f6c,-90
00116b267fd8,-73
0001e3d43a8d,-53
0001e3da0a55,-90
00147f54a4ff,-74

Figure 4.9: *Snapshot of (*left*) custom diary application and (*right*) logged ground truth with measured RSSI readings.*

Data collection was performed at common places such as the city centre (Enschede), parking lot, university campus and indoors at the office, canteen and home. In all, the spotter logs contained WLAN traces of about twelve hours duration with annotated groundtruth. Approximately 50% of the logs collected correspond to stationary phase and the remaining 50% correspond to activities performed on the move which involves walking, cycling etc. Sampling the radio environment at approximately 0.4 Hz, the twelve hour logs corresponds to roughly 16,000 samples. The logs also include different access point densities. The least number of access points in the data collected was zero, this happens when no access point is heard during a particular scan. In such cases, all our algorithms maintain the last inferred motion status until one or more access points are heard again) including the traces collected at home in several experiments.

4.6.2 Threshold learning

Each of the algorithms described in the previous section uses a certain threshold to decide whether the user's device is still or moving. As these thresholds are sensitive to several factors (e.g., environment, hardware, operating system), we use part of our data set for learning the respective thresholds and the remaining part of our data set for determining the classification accuracy using the learnt threshold. We use five-fold cross validation, where a data set is partitioned into five folds and five training and testing iterations are performed. On each iteration, four folds are used as a training set and one fold is used as a testing set.

To illustrate our thresholding scheme, let us consider finding the right threshold for the LAFC metric described in Section 4.5.2. The threshold is derived automatically from a training data set using the following method.

1. For each observation window in the training set, the LAFC is calculated.

2. Then, the distributions for both classes ("still" and "moving") are determined. Figure 4.10 (a) shows an example distribution histogram. If a threshold is applied anywhere on the LAFC axis, typically some of the "moving" observations will lie on the "still" side (in this case the right hand side) and some of the "still" observations will lie on the "moving" side (in this case the left hand side); these are the *false negatives* and *false positives*.

3. Our method now places the threshold at a position where the weighted sum of false positives and false negatives is minimal. Figure 4.11 (a) shows the amount of false positives and false negatives as well as their weighted sum as a function of the threshold. For the LAFC metric, values below (to the left of) the threshold are classified as "moving" and values above (to the right of) the threshold as "still".

   We can see that for this particular part of the data set the best threshold is 1.14 yielding a total classification error of about 9% for the training set.

4. We then use the same threshold to calculate the classification accuracy for the remaining part of the data set (i.e. the test set).

It is interesting to see how much the threshold varies among different folds that were used in a specific testing and training fold. For purposes of illustration, we only represent the extreme values in the threshold that we observed among the five different folds. Figure 4.13 (a) and Figure 4.13 (b) provides a snapshot of how large the threshold can vary among different training and test sets and the corresponding distributions in Figure 4.12 (a) and Figure 4.12 (b) respectively. We look into how these thresholds impact the final accuracy in Section 4.6.3.

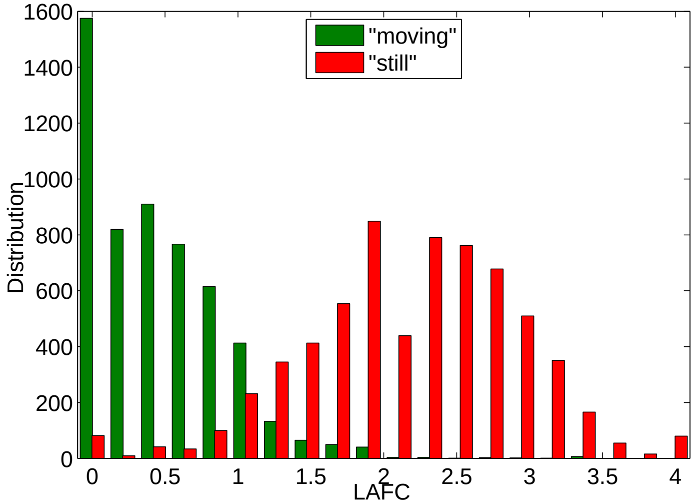


Figure 4.10: *Distribution of low-amplitude-frequency count (LAFC-Fold 4) for "still" and "moving" classes for a typical training data set (containing more than 12,000 samples in total).*

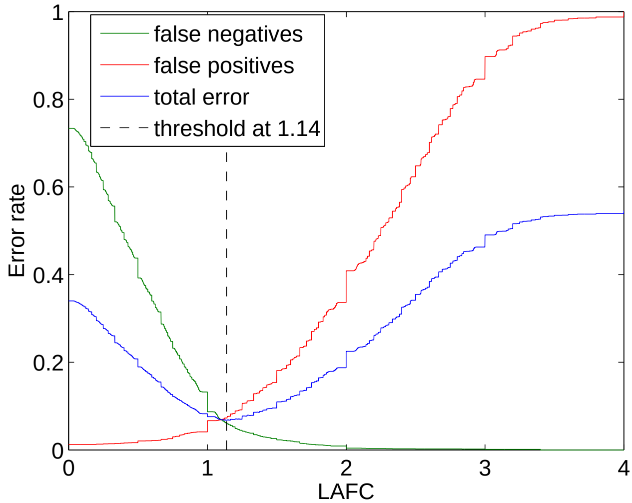
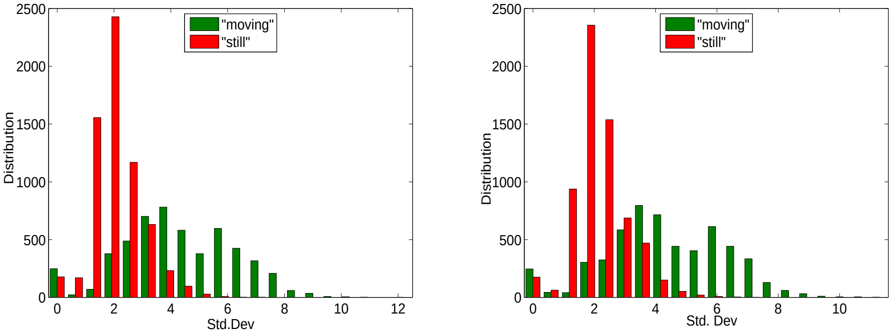
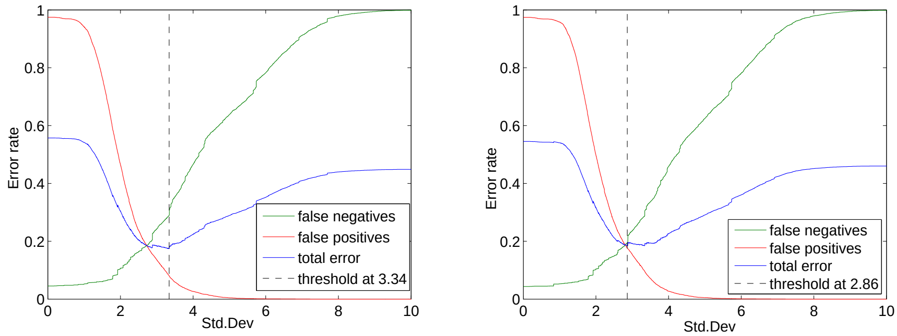


Figure 4.11: *The amount of false positives ("still" classified as "moving") and false negatives ("moving" classified as "still") as well as their weighted sum as a function of the threshold for the LAFC metric (LAFC-Fold 4).*



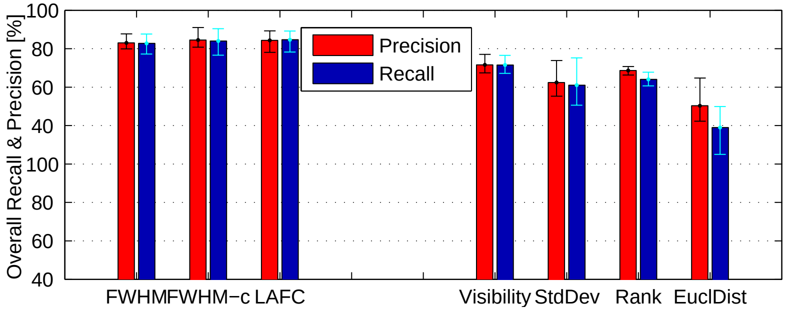
(a) Std Dev- Fold 2 (b) Std Dev- Fold 3

Figure 4.12: *Distribution of standard deviation (Std Dev) for "still" and "moving" classes for a typical training data set (containing more than 12,000 samples in total). Difference in the extreme folds are only shown here.*

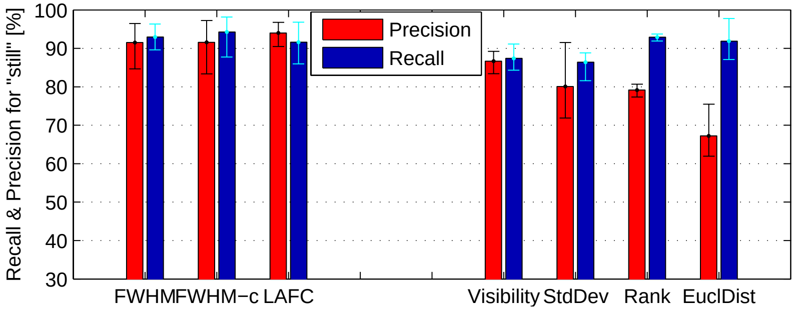


(a) Std Dev- Fold 2 (b) Std Dev- Fold 3

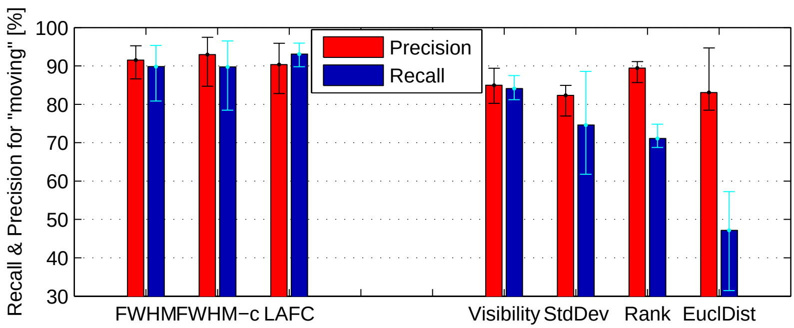
Figure 4.13: *The amount of false positives ("still" classified as "moving") and false negatives ("moving" classified as "still") as well as their weighted sum as a function of the threshold for the Std Dev metric. Std Dev fold 2 has a threshold of 3.34, while Std Dev fold 3 has a threshold of 2.86.*



(a) Comparison of the overall precision and recall



(b) Comparison of the precision and recall for still



(c) Comparison of the precision and recall for moving

Figure 4.14: *Performance of motion detection algorithms achieved by three frequency domain and four time domain algorithms: FWHM, FWHM-count, Low FFT, AP Visibility, StdDev, Rank, Euc. dist. over twelve hours of WLAN traces collected. The error bars indicate the variations in the accuracy depending on which training and test sets were used in each iteration.*

4.6.3 Results and Discussion

In this subsection, we evaluate how accurately the presented *time domain* and *frequency domain* algorithms can differentiate between moving and still states. Figure 4.14 shows an one-to-one comparison in the results obtained from both time and frequency domain algorithms tested against the same data sets.

In order to characterise thoroughly the classification performance, we use the metrics *precision* and *recall*. *Precision* for a class is defined in Eq 4.4 and *recall* in this context is defined as in Eq 4.5:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}} \tag{4.4}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive + False Negative}} \tag{4.5}$$

Figure 4.14 shows the precision and recall of the seven algorithms that we discussed in Section 4.5, using five-fold cross validation for the selection of training and test data. The classification results are averaged together for getting the final result.

The error bar in Figure 4.14 indicates the variations in the accuracy depending on which training and test sets were used in each iteration. Relating to the difference in the thresholds that we have observed among different folds as shown in Figure 4.13 (a) and Figure 4.13 (b), we observe the stronger variation in the error bar pertaining to the Std Dev metric in Figure 4.14. However, looking at the LAFC metric, where we did not observe any difference among each fold and error bars, still shows a variability in the accuracy. This highlights that the sensitivity of a particular algorithm does not depend only on the variations in the thresholds observed among different folds, but also depend on the underlying data itself.

Looking at the results leads to the following conclusions. The performance of the frequency domain algorithms show a better precision and recall when compared to the time domain algorithms. Low precision indicates that many false positives exist and a lower recall indicates that many groundtruth events were missed. Among the different time domain algorithms used, the Euclidean distance based algorithm performs the worst, while the remaining three algorithms perform about the same. It is interesting to note that a simple count of the number of access points seen achieves an overall accuracy of 85%. The accuracy obtained is similar to the the accuracy reported by Krumm et al. [110] using an algorithm based on the temporal variation of RSSI. Of course it is hard to make an one-to-one comparison among algorithms used here and those available in literature, as the data used for testing is different in both cases. But since we have performed the evaluation for data collected from different environments and settings, we do not expect a drastic difference in the performance, when operating on data taken in similar environments.

Comparing the different frequency domain algorithms, all the three algorithms – FWHM, FWHM-count and LAFC based algorithms results are comparable. Another important aspect we observed is that generally the thresholds for the frequency domain metrics are not as sensitive to external influences as the ones for the time domain metrics. Also, some of the time domain metrics are particularly sensitive (e.g. standard deviation (Figure 4.7) and Euclidean distance (Figure 4.8)).

The overall classification accuracy obtained with time domain algorithm (about 85% with AP visibility, Std Deviation and Rank) is comparable to the one reported by Sohn et al. [174](85%). Although, Sohn et al. achieved this accuracy for a three state classification scheme and our results are for two state classification, it is interesting to note that all our presented algorithms use only one feature as opposed to Sohn's work where seven different features were used to train and test data. Employing other features, as reported in Sohn et al. might result in a significant improvement in the accuracy. This is yet to be investigated.

Investigating how well the algorithms can cope in identifying the transitions between still and motion states, we have observed that it typically takes half an observation window for any metric to cross its threshold between states. This is of course not surprising, because immediately after the transition, the "old motion state" samples will still occupy the major part of the observation window. Only after half of the observation window has actually passed the transition, the majority of its samples will be "new motion state" samples. We therefore interpret a classification at time t as the estimated motion status for time $t - \frac{1}{2}T_{\text{windowsize}}$ where $T_{\text{windowsize}}$ is the length of the observation window.

Frequency domain algorithms, specifically the FWHM count algorithm, performs very well for all experimental settings by achieving an overall classification accuracy of 92%, clearly outperforming all the other motion inference algorithms. It is thus reasonable to conclude that the scheme's detection accuracy and performance is significant after being tested across various settings. Fine tuning the thresholds and/or incorporating more features together might even further increase the accuracy. The results show that we are able to distinguish between still and moving states with a high accuracy without having to instrument a person with any additional sensors. It is interesting to consider the performance of the presented algorithms for different window sizes, access point densities and FFT lengths.

Effect of window size

Changing the length of the observation window impacts the results of all the presented algorithms. The results are shown in Figure 4.15. In general, higher the window size, the better the performance. However, it is important to note that higher window sizes also increases the latency. Given the sampling rate of 0.4 Hz, changing the window size from two to sixteen corresponds to a wait time of 2.5 to 20 s (i.e. half the observation window's duration).

Effect of access point density

We took three non-overlapping subsets from the total log in order to test the sensitivity of the various metrics' learnt thresholds to the access point density. One subset contained measurements collected at areas with lower access point density such as a home environment (typically around four access points) and in some shopping areas outside of the city with just one or two access points. On average this "sparse test set" had three access points after removing the weaker access points. The second subset contained measurements collected in denser access point environments such as the office. On average this "dense test set" had nine access points after removing the weaker access points. The third subset was represen-

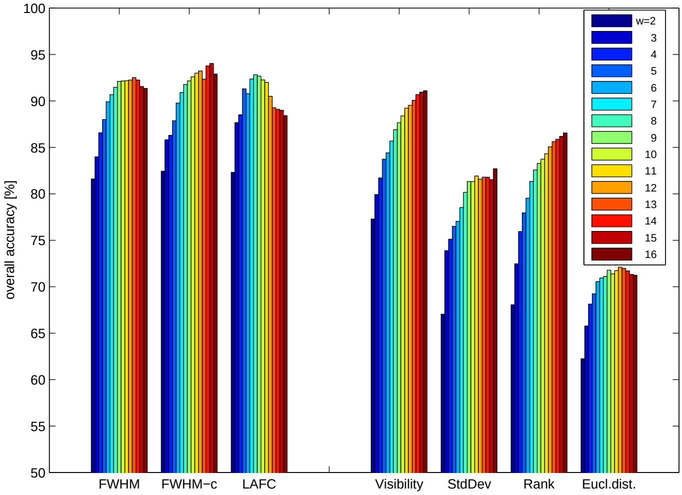


Figure 4.15: *Effect of different window sizes, two to sixteen.*

tative for the total log, containing measurements from a variety of environments. We used this "training set" to obtain the thresholds for the various metrics, as explained earlier in Section 4.6.2. These thresholds were then used in the evaluation of the metrics for both the "sparse" test set and the "dense" one.

The results are shown in Figure 4.16. It is clear that the more APs are seen, the better the motion status classification performs. In particular, the learnt thresholds for time domain metrics are very sensitive to this factor.

Effect of n- point FFT

Figure 4.17 shows that increasing N in an N-point FFT does not improve the accuracy results for the frequency domain metrics for $N > 16$. Therefore we can safely use an N-point FFT with a value of $N = 16$ (for FWHM and FWHM-count) or $N = 8$ (for LAFC) in order to keep the computational cost low.

4.7 Localisation

In this section, we outline the location algorithms that operate on RSSI measurements. The goal of our work is to demonstrate that algorithms which rely on both (i) the location of access points and (ii) a coarse estimate of the relative distance to the access points can benefit from adding motion information that we presented earlier in Section 4.5. Many of the probabilistic approaches like the particle or Kalman filtering use an inherent motion model to enable localisation and tracking. These algorithms work in a "predictor-corrector" fashion, by weighting the filter model more heavily when the errors in the raw measurement go higher, thereby making the final estimates more accurate. We have used a similar approach

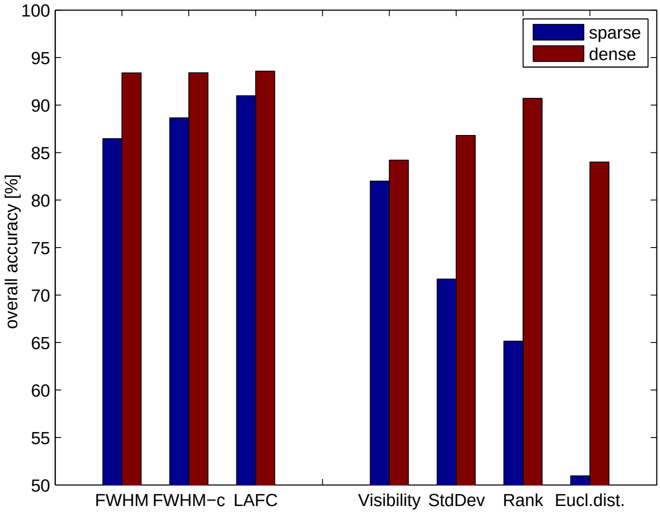


Figure 4.16: *Effect of AP density– Sparse vs. Dense (sparse contains an average of three AP and dense contains an average of nine AP).*

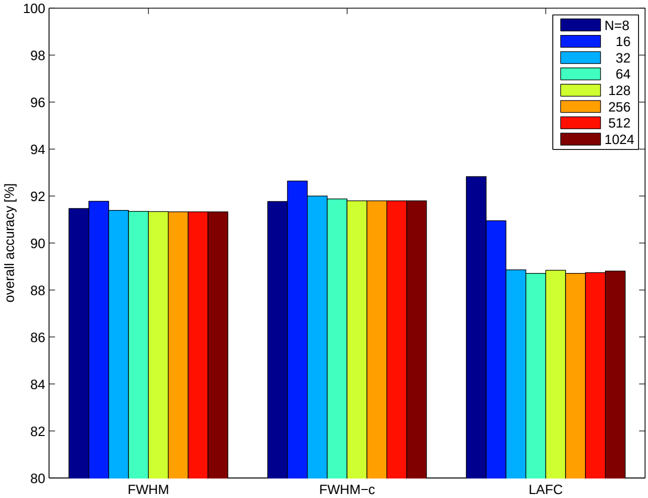


Figure 4.17: *Effect of FFT width for the frequency domain metrics.*

but instead coupled to variants of the simple centroid algorithm.

WLAN-based location systems predominantly use received signal strength measured at the mobile device to estimate location. In general, signal samples fluctuate even in static environments. This results in unstable location estimation even while the user or the device is still. In order to overcome errors due to instability in RSSI used for location estimation, most positioning algorithms use some kind of time averaging to smoothen the RSSI values and perform temporal smoothing on final estimates. In many cases temporal smoothing of several location readings can result in better accuracy. However, the length of the history of measurements to be used for temporal smoothing is to be chosen carefully. A smaller history size would result in jumping of location readings taken from one reading to the next reading can move large distances. This can be characterised as a "teleportation" effect. This jumpiness will often result in an unpleasant experience to the users if they were to use an application such as a buddy finder or viewing their location on a map on their mobile devices. One way is to use the model of the building layout as part of the location algorithm so that unrealistic jumps through walls and other obstructions, or unreasonably from one floor to another floor in the environment could be avoided.

We incorporate the knowledge about the user state, i.e. moving or still, as a part of localisation algorithm itself. As an example, when the user state is deduced as still, all the unwanted jumps caused by the location estimation can be ignored. When the user state is deduced as moving, an appropriate motion model can be added as a part of the location algorithm. Therefore after each location update the new location estimate can move by a maximum distance depending on user's speed. In this way, many false location estimations can be curtailed, leading to a better accuracy and resulting in location traces which are smoother.

4.7.1 Related Work

Localisation using WLAN-RSSI can be divided into modelling-based and radio mapping-based approach. In the former, the user's location is calculated from a radio propagation model and beacon locations. In the radio mapping approach, a radio map of the environment if constructed during an offline phase and the radio map is used to locate the user during an online phase. RADAR [33] is one of the earliest WLAN-based location systems. Since then, there has been a lot of research performed around this topic. RADAR operates by recording signal strength information from multiple base stations. It takes advantage of two observations–firstly, the signal strength varies considerably spatially (even at smaller distances); secondly, the signal shows a temporal stability (when the user's device is held stationary). Using these two observations, Bahl et al. proposed a technique called "radio fingerprinting". Two phases are involved in radio fingerprinting based system–*offline phase* (i.e. collection of fingerprints) and *online phase*, where the actual location of the device is estimated based on the RSSI measurements of access points and the knowledge of the fingerprints collected during the offline phase. Obviously, depending on the density of the collected fingerprints the accuracy of the system varies. There are two disadvantages with this approach–first, is the lack of scalability of this technique and secondly, if an access point in the environment is to be moved to a different place, the collected fingerprints are not valid anymore and fingerprints have to be frequently calibrated to the changing environment.

The users' location can also be computed by probabilistic approaches [46] or a neural network model [172] as opposed to the deterministic approach explained before. Joint clustering [203] and Bayesian networks [113] are similar to RADAR. They all use a training session to get many fingerprints and using them they try to predict the location. The Ekahau positioning system [56] is a commercial system which is able to locate clients and provides the coordinates in 3D corresponding to each client. The main positioning module is run on the server. They report an accuracy of about 1–2 m, however it requires a considerable calibration effort.

Horus [202] is a fingerprinting-based location system that identifies different causes of variations of the signal strength (e.g. handling small scale variations, correlations between successive samples etc.) and incorporates various techniques to handle them. It has been demonstrated to work better than normal fingerprinting-based methods [33].

However, radio map-based methods are time-consuming to construct and have to be updated should there be a change or addition in the access points location. As a result many deployments have been small, or restricted to the corridors of larger buildings. In one of the few large-scale deployments of an RF-based location system, twenty-eight person-hours were required to construct a radio map covering a 12,000 m^2 building [77]. Robots have been used to acquire radio maps for use in robot location system. However, such a map would not be well suited for use in a human location system for tracking people, since it would not take the presence of the user's body into account.

Contrary to the radio map-based approach which requires intensive calibration effort, modelling-based approaches have been used in several projects, most notably in Active campus [1] and Placelab [115]. Active campus [1, 75] project goals were to offer location-based social networking services to students and was the first wide area deployment of WLAN location system. Active Campus uses a propagation modelling-based approach, as it is not feasible to collect fingerprints over a wider area (i.e. campus-wide) and the authors have reported an average estimation accuracy of 22 m outdoors and 11 m indoors. Placelab [115], developed by Intel, allows commodity hardware clients like laptops, PDAs and cell phones to locate themselves by scanning for radio beacons such as 802.11 access points, GSM cell towers and fixed Bluetooth devices. It does not involve much calibration, as information about the access points and GSM cell towers are collected through *war driving*. It has been demonstrated only for outdoor environments, reporting an accuracy of 13 to 20 m [116]. It maintains privacy by computing the locations of the users at the client device. Placelab does not provide a mechanism to share locations. However, it is being integrated into different systems such as Active Campus [76], Google's Latitude [72] providing that functionality. For WLAN localisation using time-based techniques refer to [49].

4.7.2 Smoothing location estimates by incorporating movement model

The positioning functionality is implemented using *filter chains*, which represent a sequence of calculations performed on the observed RSSI measurements. We use the spotter library from Placelab [12] to record RSSI from neighboring access points for devices like normal PC and laptops. To run on Windows Mobile platform, the spotter functionality from the OpenNETCF.Net.dll library has been used. We rely on the location of the access points mapped in 3D and other information pertaining to the access points such as the MAC addresses and the transmit power setting of the access point. The topology of the access points could either be managed as a database residing in the network, or it could be a local configuration file in order to minimise the network dependencies and maintain privacy. Obtaining this information is easier in a university setting as they typically maintain a database of network access points. In the literature, there exists many ways such as war driving or stumbling to obtain the neighboring access point coordinates [114].

Moving average

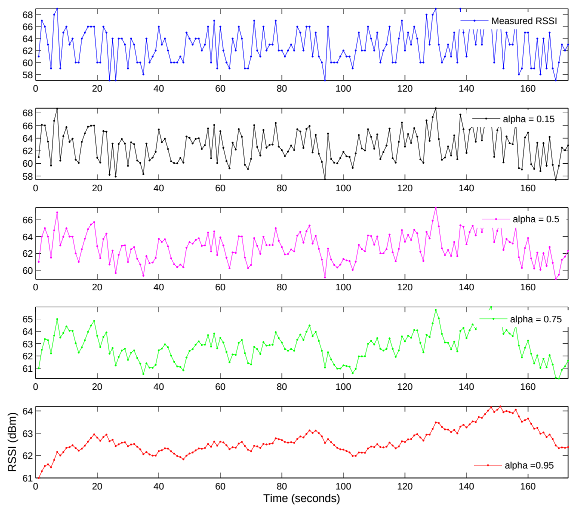


Figure 4.18: *Effect of α on filtering RSSI. Observing the RSSI values of each subplots, it is clear that a higher value of α shows extreme averaging effect compared to lower values of α.*

We use an exponential moving-average filter to reduce the effect caused due to the noise and smoothen the received signal strength for the analysis presented here. Equation 4.6 shows the formula.

$$\mathrm{RSSI}_{filtered} = \alpha \times \mathrm{RSSI}_{prev} + (1 - \alpha) \times \mathrm{RSSI}_{current} \tag{4.6}$$

Equation 4.6 states the current signal strength $\mathrm{RSSI}_{filtered}$ value is a linear aggregate of the previous signal strength RSSI_{prev}, $\mathrm{RSSI}_{current}$ values and an independent weighting factor (α). The weighting for each of the older observations decreases exponentially, giving much more priority to recent observations while still not discarding the older observations entirely. A lower α discounts older observations faster. The degree of weighting is expressed as a constant smoothing factor α, a number between 0 and 1. α may be expressed as a percentage, so a smoothing factor of 15% is equivalent to $\alpha = 0.15$. Moving averaging has been used in other related work in the area of WLAN localisation [202]. Figure 4.18 illustrates the effect of different values of α on RSSI observed in signal readings from one access point. We use an $\alpha = 0.2$. Of course the challenge is how to determine what exact α is required. We briefly discuss in Section 4.8, the impact does choosing a proper α on localisation accuracy.

Localisation

The core positioning algorithm is based on a weighted centroid approach. The difference between the normal centroid and weighted centroid is that the latter introduces variable weights for each access point. Weighted centroid uses the relative distance estimates to the strongest access points. This is performed by assigning the location of each of the several strongest access points a weight in the position calculation based on the relative distance between those estimates. Obtaining absolute distances precisely between the access points and the mobile device to be located is harder due to the multipath reflections that are predominant in indoor environments. Hence algorithms that make use of absolute distances retrieved from WLAN RSSI, such as those based on trilateration perform poorly, especially when the access points are arranged in a collinear fashion.

We estimate the *relative distances* based on the transmit power of the access points that are available and the RSSI values which typically correspond to the power at the receiving end. Although the distance estimates are not accurate, it will give a clue to which access points are closer and hence are to be used in the position estimation. Based on calculating the approximate *relative distances* to access points, we do not make use of the access points that are considerably farther. Alternatively, one could use thresholding based on RSSI to eliminate the redundant access points. From the RSSI, we use the motion detection algorithm that we explained earlier in Section 4.5 to detect the state of the device as either still or moving.

Using motion information

Depending on the state, two filter chains are used appropriately:

- When the motion detection algorithm returns the state of the user as moving, we employ a motion model which smoothens the final location estimates gracefully, by preventing any large movements between two different time steps. The motion model filter uses a maximum allowable distance, depending on the users walking speed within a stipulated time frame. For instance, if the walking speed of the user is 1.5 m/s, then difference in the current location reading and the next location reading cannot be greater than 3 m if we sample radio environment every 2 seconds. Essentially the filter checks if the distance moved between the time between the current and the past location is greater than this threshold limit. If the difference exceeds the set limit, the location is updated solely based on the motion model. This essentially acts like a correction phase i.e. the position estimated is corrected by the knowledge of motion information.

- When the user is still, ideally the user's estimated position must remain still at the same point. But since the signal strength varies even at static location, the estimated location often jumps even when the device is still. We use a lesser value of maximum allowable distances (i.e. a fluctuations of 0.2 m is permissible) for the static cases and use a history of measurements to average the results together.

4.7.3 Algorithms used for comparison

We compare the accuracy of the presented Weighted Centroid with motion model as described in Section 4.7.2 to that of other position estimation algorithms with and without adding motion information. In total we have four algorithms to compare with (i) Weighted Centroid (with motion model) (ii) Centroid (with motion model) (iii) Weighted centroid and (iv) Centroid algorithm.

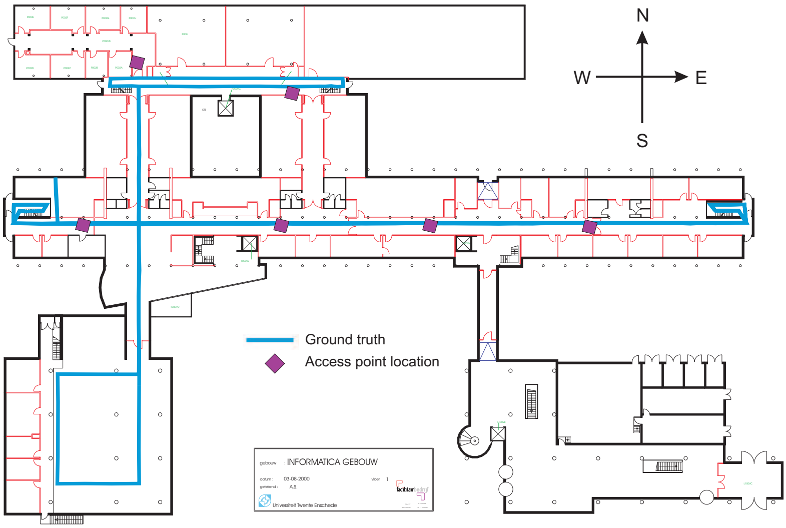
The Centroid algorithm places the user on the geometric centroid of the strongest access points that appeared on the current scan. Weighted centroid as we explained before assigns specific weights to the access points based on the estimated relative distances. Centroid with motion model essentially uses the same principle as we explained before, if the difference in the moved distance between two timestamps are greater than a certain allowed distance, it corrects the final estimate.

4.8 Experimental evaluation

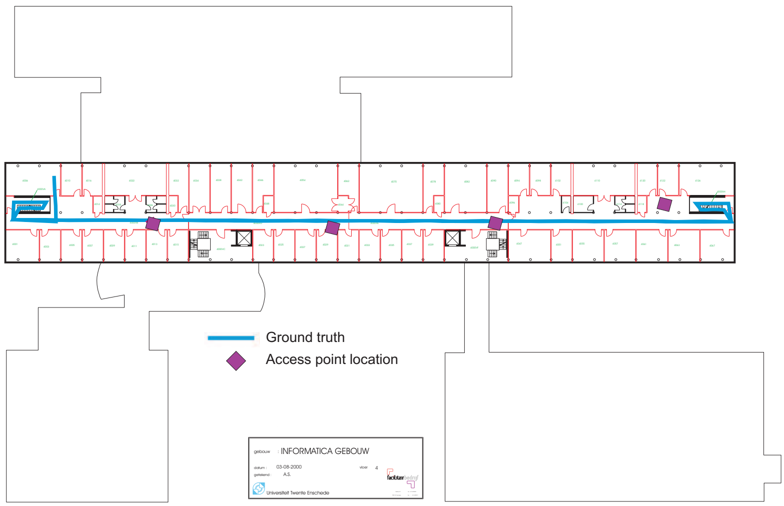
In this section, we report the performance of the location algorithms that were described in Section 4.7. First we outline in Section 4.8.1, the testbed used for collecting the WLAN RSSI traces. We then summarise the performance of the location algorithms in Section 4.8.2.

4.8.1 Data collection

All the experiments to assess localisation accuracy were performed in a five-storied, Zilverling building at University of Twente. Floor 2–5 has a dimension of 106 m $\times$ 14.5 m and have a same layout with a long corridor and many rooms and has four access points that are mounted on the ceiling and are placed in a straight line (refer Fig. 4.19b). The first floor (refer Figure 4.19a) has a different layout and spans along north-south side as well with a few additional access points covering the north and south side of the floor. The transmit power of most of the access points were either 50 mW or 30 mW. The measurements were taken from floor 4 (east side) till the floor 4 (west). Through the flight of stairs located at the extreme west, the data collector traversed to floor 3, from west end to east end. This continued until floor 1, where the data collector traversed the other hall way (north-south side) as well. Through the flight of stairs repeated the same trajectory all the way until floor 5 and finally reached the starting point on floor 4. The data collector walked at a normal walking pace of 1.3-1.5 m/s. Data was recorded as one trace lasting approximately twenty minutes, resulting in about 350 RSSI readings (with an average difference in time-step between consecutive samples of 2.5 seconds between). Out of the collected spotter measurements, fifty access points were detected in total in the entire trace and our database had only twenty-eight access points location mapped. So among those mapped twenty-eight, few of them were taken for the position estimation and the rest of the twenty-two access points were not used in the estimation process. Our sampling time was limited to the Windows Mobile platform that was running on the mobile device. Since the same data was to be tested with different algorithms, we logged the measurements and all the analyses were done offline.



(a) Floorplan of the first floor



(b) Floor plan of the $2-5^{th}$ floor

Figure 4.19: *Testbed used for collecting WLAN RSSI traces is a five-storied, Zilverling building, University of Twente.*

Groundtruth

It is rather difficult to obtain an accurate groundtruth to this scale, covering the whole building. The normal procedure would be to adopt a map clicking application which can exactly match the readings collected to the actual place. This process should work reasonably well but with small amount of error when used on a laptop. Since we used the smaller device the granularity of the recorded location would have been worse because of the limited screen size on the device. Hence, we followed a slightly different approach. The user was logging measurements using the same diary application as used for collecting RSSI traces for evaluating the motion detection algorithm. At crucial points in the path, such as corners and stairs, the motion status was recorded so as to log insertion points where we manually inserted the corresponding positions and we used an interpolation script to interpolate based on the time-stamps in the actual measurement log the corresponding distance moved.

4.8.2 Results and Discussion

In this section, we report the performance of the four localisation algorithms - (i) Centroid (ii) Weighed centroid (iii) Centroid with motion model (iv) Weighted centroid with motion model. Table 4.2 and Figure 4.20 summarise the overall results of the algorithms.

Without adding motion, the two algorithms (Centroid and Weighted Centroid) report median accuracies of 8.41 m and 6.87 m respectively. Adding motion results, yields median accuracies of 7.49 m and 5.14 m respectively. This is because the motion model filter utilises its predicted estimate for the position of the device, in addition to estimates calculated using current RSSI observations, to produce the new estimate (similar to that of a Kalman filter). The cumulative distribution shows even the seventy-fifth percentile error reports less than 9 m error for weighted centroid with motion incorporated, given the fact that we have completely avoided the intensive radio-mapping process which is typically used in fingerprinting algorithms. Figure 4.23 reveals that a considerable portion of the error occurs when the user is at the extreme end of the corridor, as typically the extreme ends have much lower access point density. Note that the building has no nearby neighboring buildings and hence the access points which appear are only those from the building. Also looking at Figure 4.23 the first floor measurements do not report any location estimates in the south side of the floor. This is again due to the unavailability of the access points in that region (as seen in Figure 4.19a). Considering the fact that 15% of the readings constitute either stair cases or at the south side of floor 1, the reported mean accuracy of 6.68 m for the Weighted centroid with motion is reasonable for a calibration-free approach.

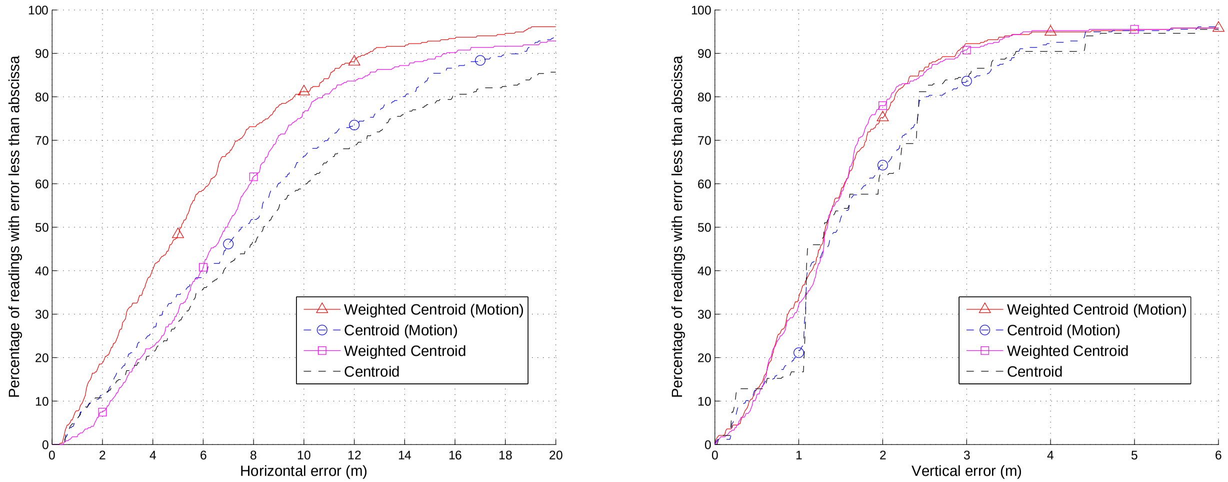


Figure 4.20: *Cumulative distribution comparing the accuracy of four presented algorithms tested with a twenty minute trace of RSSI measurements.*

Algorithm	Mean hor. error	Mean vert. error	50% conf. level (m) Horizontal	Vertical	75% conf. level (m) Horizontal	Vertical
Centroid	10.77	1.73	8.41	1.30	13.48	2.43
Weighted Centroid	8.53	1.51	6.87	1.34	9.74	1.84
Centroid (with motion)	9.01	1.84	7.49	1.47	12.47	2.42
Weighted Centroid (+ motion)	6.68	1.53	5.14	1.32	8.57	1.98

Table 4.2: *Tracking performance summary. All values shown pertain to the location results of the "walking" data traces collected for about twenty minutes in a five-storied building, covering all five floors during the measurement period.*

We mentioned in Section 4.7.2 the consequences of choosing the right the weighting factor (α) on localisation accuracy. Figure 4.18 shows how the signal variations reduce when a higher value of α is used. This is particularly good when the device is still, and we would want a clean signal to be able to choose the access point for estimating the location. However, this extreme smoothing of signal will not help much for the case when the device is moving. When we increased the α to 0.9, we noticed that the mean error in the weighted centroid algorithm with motion increased from 6.68 m to 9.84 m. This holds the same for the case when a larger window of history size is used for temporal smoothing of position estimates. We observed by raising the history window from five samples to twenty samples, increased the error from 6.84 m to nearly 20 m. This is because adding a history size of that length, contributes to a considerable delay in estimating the position and thereby severely degrading the accuracy. We believe our presented motion detection schemes can be applied to adapt both the weighting factor and the history window and thereby lead to a considerable improvement in location accuracy. Automatically tuning these thresholds is part of our future work.

Floor Identification:

This subsection reports how the presented algorithms detect correct floor information. This is very important for many of the applications to identify at which floor a user is present. Figure 4.22 shows a plot of the error in floor estimates for the Weighted centroid with motion and the Table 4.3 gives the percentage of the measurement time, each algorithm reporting that the user is in the correct floor. For most of the measurements, the algorithm reports that the user is in the correct floor. It is particularly interesting to note that for periods between 125–150 samples, the error becomes worse. These measurement samples refers to the south side measurement done at floor 1, where we did not have any access points mapped, hence any access points heard at that point were from higher floors, thereby pulling the floor estimates higher by two floors. Table 4.3 and Figure 4.21 summarise the error in floor estimates reported by all four algorithms on a per-floor basis. It is clear that the error in the first floor contributes to the maximum error, as all algorithms consistently show an average of only 50% correct estimation. Excluding the first floor measurements, in principle, we can show that we are able to identify the correct floor around 82% of the measurement time. Table 4.3 shows that there is a modest improvement between the algorithms that use motion information over the other algorithms without motion model.

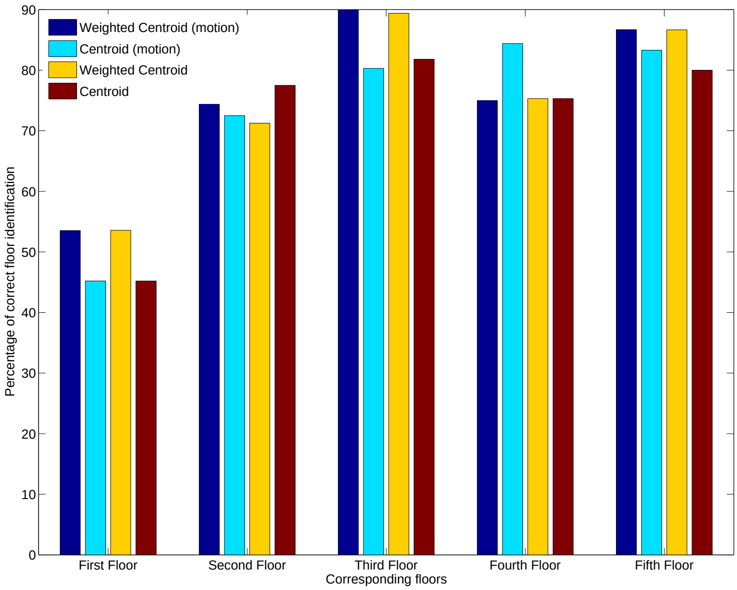


Figure 4.21: *Accuracy in floor estimation represented as per floor achieved accuracy.*

Algorithm	All floors	Floors 2–5	Accuracy per floor Floor 1	Floor 2	Floor 3	Floor 4	Floor 5
Centroid	70.9	79.4	45.2	72.5	80.3	84.4	83.3
Weighted Centroid	70.0	78.3	45.2	77.5	81.8	75.3	80.0
Centroid (with motion)	72.7	79.5	53.6	71.3	89.4	75.3	86.7
Weighted Centroid (+ motion)	75.1	82.2	53.6	80.0	90.0	75.0	86.0

Table 4.3: *Accuracy of floor estimation, represented on a per-floor basis (percentages).*

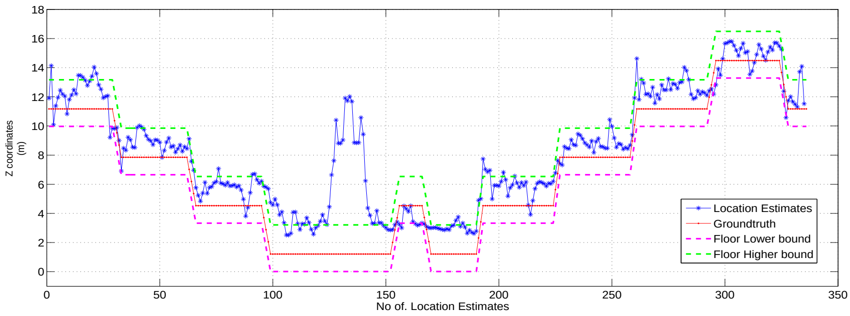


Figure 4.22: *Floor identification error. The peak between 125-150 s is due to the unavailability of the access points in the floor where the measurement was performed.*

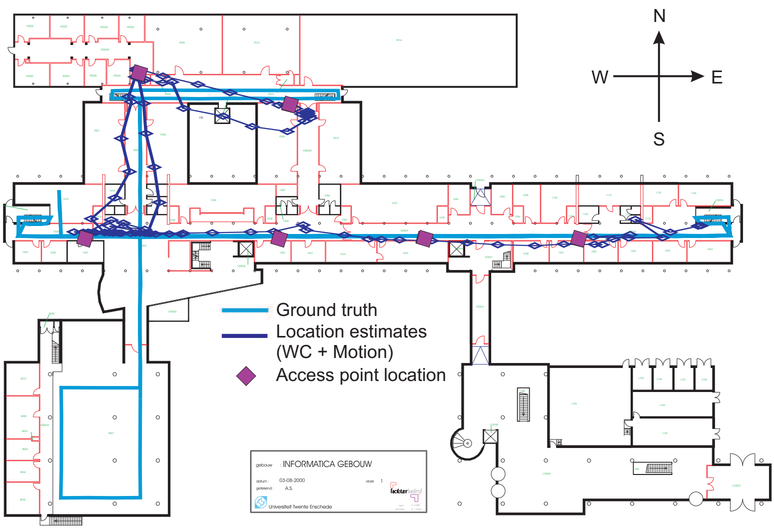


Figure 4.23: *Estimated position overlaid on a floor plan (groundtruth and the location estimates pertain to floor 1 measurements only). Comparing the trajectory of the groundtruth and estimated location shows the error mostly happens on the stairs and towards the extreme end of the corridor.*

4.9 Architecture for sharing location

The algorithms presented in the previous part of this chapter were used in a prototype called FLAVOUR, which was originally designed to serve as a conference assistant. Figure 4.24 shows a high-level view of the service-oriented architecture used for sharing location in FLAVOUR. All the services can be discovered through the *Lookup Service*. The information about the access points is stored locally in a database, which is used by FLAVOUR to compute the location of the users. Another source of information to be used is the topology that is stored as geo-referenced maps. Using these maps, the users are able to visualise their location, and the location of other people.

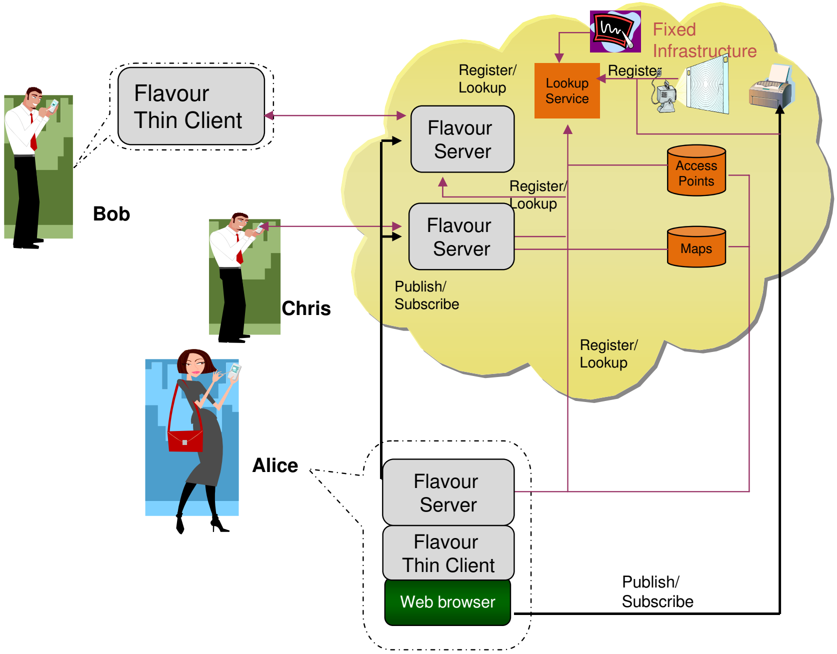


Figure 4.24: *High level view of the system architecture.*

We have created a thin client that has to run on the users mobile device, while the server part can either run on the mobile device or on the "infrastructure". By infrastructure we mean a server which does not need to be switched off because it is running on batteries and has a permanent network connection. The advantage of running the *LocationServer* on the infrastructure is that the location service can still be provided even if the client's device is off. The time-stamped location provided will be the last location in which the user's device was on. Furthermore, in this manner nothing can be concluded from the existence of the user's location in the *Lookup service*, because his location service is always available. On the other hand, the users may not trust the system or may not be interested in sharing their location when they are off-line, and, thus, run the *LocationServer* on the mobile device. At present we cannot guarantee absolute privacy of the data when the *LocationServer* runs on the infrastructure, as in principle it is possible that the administrator of the system hosting the software 'spies' on the user.

As shown in Figure 4.25 the functionality provided by FLAVOUR can be divided into:

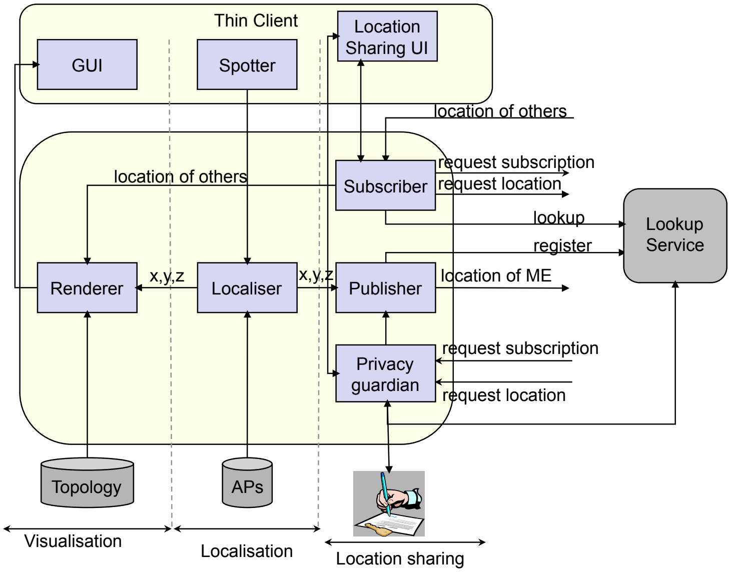
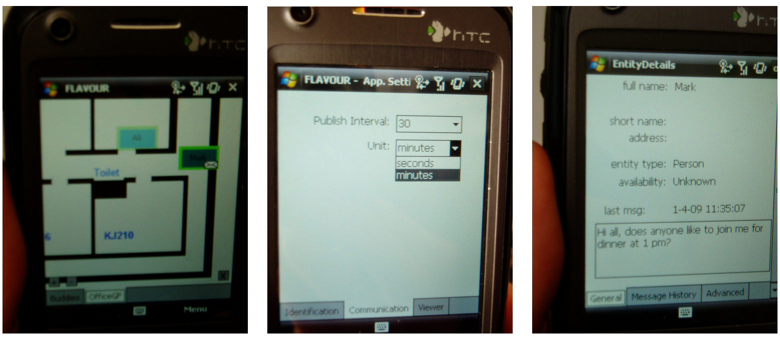


Figure 4.25: *Components of FLAVOUR.*



(a) Viewing Buddy location (b) Publishing location (c) Messaging

Figure 4.26: *Snapshot of FLAVOUR.*

- **Location Sharing:** The users can provide their location to other users and can be aware of other user's location. There are two mechanisms to share location information: *publish/subscribe* and *request*. With the former the publisher sends updates whenever the location of the user changes in a significant manner, while with the latter the information is provided as a reply to a (one time) request. In both cases the *Privacy Guardian* decides if the request should be rejected or accepted, and if accepted under which conditions (e.g. granularity, update frequency, duration). The location sharing functionality is discussed in more detail below.

- **Localisation:** In order to determine the user's location the *Spotter* measures the signal strength of all the access points it hears. The *Spotter* sends those measurements to the *Localiser*, which in turn will use them to compute the user's location.
- **Visualisation:** The users can visualise their location, as well as the location of other users and points of interest on a map using the *Map Viewer*. The *Renderer* composes maps using the topology of the venue and points of interest, the coordinates of the user provided by the *Localiser*, and the location of other users provided by the *Location Subscriber*. Figure 4.26 illustrates a snapshot of the FLAVOUR interface.

Location Sharing

A very important issue for sharing location is privacy. In FLAVOUR we want each user to decide what location information can be disclosed and to control when and how it is disclosed. The *Lookup service* provides an entry for every person that registers to the conference. The information in the *Lookup service* is the user's name, affiliation and a pointer to his *LocationServer*. The basic interface provided by FLAVOUR to other users is:

- getLocation(Requester, Reason): One time request for the location of the user.
- subscribeToLocation(Requester, Reason): Subscribe permanently (i.e. during the conference) to the location of the user.
- subscribeToLocation(Requester, Reason, TimePeriod): Subscribe during a certain period to the location of the user for a given period.

The reply to a request for subscription may restrict the subscription to a shorter time period and additionally put restrictions to the accuracy of the provided location and frequency of the updates. In all cases the request may be rejected.

The *Privacy Guardian* uses the identity of the *Requester* to decide if the service requested should be provided and with what restrictions. On the one hand, the *Privacy Guardian* checks the identity of the *Requester* (if he is who he says) using a challenge-response protocol. For this purpose it uses the *public key authority* service provided by the system that stores the public keys of the users. In order to decide who can access the services and under which conditions, the user can create a buddy list using the *Location Sharing User Interface* and assigning access rights. If a request comes in from a person not in the list, the *Privacy Guardian* can ask the user for action through the *Location Sharing UI* using a similar mechanism to a cookie blocker that provides options as *'allow once'*, *'allow during conference'*, *'block once'*, *'block always'*, or set explicit time restrictions and intervals for updates. In the future we will add the capacity to set the accuracy of the location provided as well. The *Privacy Guardian* stores the reply from the user to decide what to do next time a request from the same person arrives.

All arriving requests and their replies are logged by the *Privacy Guardian*. In this way the user can afterwards analyse who requested his location and why by looking at the provided

Reason. At present the *Privacy Guardian* does not have the capacity to analyse the given *Reason*, instead it either just passes it on to the user to decide to accept or reject the request, or simply logs it for future analysis. Thus, we rely on normal social control to prevent abuse of the system (see [121] for a discussion on the subject). The *Privacy Guardian* also has tools to let the user analyse the log and to provide warnings in case of possible abuses, for instance if a user inquires for other user's location very frequently. The frequent inquiring may be justified by the given *Reason*, otherwise the user may find frequent inquiring a breach of trust and consequently revokes or restricts the requester's access rights.

Allowing individual requests is more privacy preservant than allowing subscriptions. As by allowing subscriptions the user cannot see when the subscriber is really looking at his location. Thus, basically he is allowing to be tracked. Although the study performed on Active Campus by Griswold et al. [76] shows that users are not bothered by permanently sharing their location with their friends, we believe that allowing one-time requests may be more desirable. When a request is performed the user does not need to be immediately notified (and since they may be bothered by this fact), especially if he has authorised the requester to have access to his location at any moment. However, the request is logged for accountability purposes.

4.10 Conclusions

This chapter addresses how to sense motion and location, leveraging existing WLAN infrastructure. Based on the characterisation of the reported RSSI measurements we developed a range of motion detection algorithms to detect still and moving states. We identified a rich set of features that could be gathered based on either temporal or spectral characterisation. Our motion detection algorithm, exploiting the frequency domain characteristics yields a precision and recall of over 90%. We also showed that the sensitivity of a particular algorithm was not only dependent on the chosen threshold, but also depends on the underlying data. It will be interesting to consider the resource requirements of each the algorithms to analyse the tradeoff between accuracy and complexity of the presented algorithms. One possibility of extending this work is to use a combined set of features in a machine learning algorithm, to obtain finer accuracy, and also explore the possibility of identifying more states like "cycling" or "driving".

We also showed the benefit of combining motion information with the location algorithm. A median error of approximately 5 m can be achieved without the use of calibration. We have based our analysis by testing the algorithms on a typical set up used in many office environments, where access points are arranged linearly. However these results cannot be generalised, as the results are very much dependent on the density and topology of the access points in the test area.

The improvements in the algorithms with motion incorporated suggests that many of the calibration intensive fingerprinting algorithms could use such a scheme for detecting users in areas such as hallways, and restrict the fingerprints only to the rooms, as we envisage that our method might not work as well there. This again depends on the access point configuration in the test area. If there are access points also distributed along spatially seperated axis it will result in considerable improvement because it will allow for better trilateration. In general, if the access points are deployed not only to provide good coverage (i.e. useful for communication purposes), but if they are deployed keeping in mind that such infrastructures can be used for positioning purposes, we can expect much more improvements.

When incorporating the motion information we assumed the walking speed of the user is known, usage of other sensors which can actually give us the speed and direction, for instance by using a combination of accelerometers, gyroscopes and magnetometer could enhance the accuracy further. Other possible directions of research includes, incorporating map-matching methods and in combination with probabilistic methods like particle or Kalman filtering.

We also showed in brief how motion detection algorithms can be applied to give an indication of what degree of history size can be used for temporal smoothing of the location estimates (i.e. adaptive windowing) e.g. by setting a higher history size, when still and lower, when moving. The weighting factor (α) to be used in the moving averaging can also be automated (i.e. adaptive thresholding) e.g. choosing higher α when the device is still and smaller α when the device is moving. Although addressed in short, we showed what scale of improvements can automatic tuning of these thresholds can bring.

Finally, we have presented an architecture to share the location among peers. Following Langhereich's guidelines [119] we have made privacy an important design consideration of our architecture. There are two main design issues that make FLAVOUR privacy-observant. On one hand, it adheres to the widely accepted notion of privacy formulated by Westin in 1967 that "privacy is the claim of individuals to determine for themselves when, how, and to what extent information about them is communicated to others" [197]. On the other hand, the location is determined by software controlled by the users, which either runs completely on their mobile devices or partly on their mobile devices and partly on the infrastructure. Thus, there are no centralised services that are solely responsible for tracking the users' location.

CHAPTER V *

Ultra-wideband positioning using pseudoranges and angle-of-arrival

This chapter presents two algorithms, non-linear regression and Kalman filtering, that fuse heterogeneous data (pseudorange and angle-of-arrival) from an ultra-wideband positioning system. The performance of both the algorithms is evaluated using real data from two deployments, for both static and dynamic scenarios. We also consider the effectiveness of the proposed algorithms for systems with reduced infrastructure (lower deployment density), and for lower-complexity sensing platforms which are only capable of providing either pseudorange or angle-of-arrival.

*This chapter is a minor revision of the paper published with the title"Position Estimation from UWB Pseudorange and Angle-of-Arrival: A Comparison of Non-linear Regression and Kalman Filtering " In Proceedings of Location and Context Awareness (LoCA), Tokyo, Japan, May 2009 [137].

5.1 Introduction

Location systems based on conventional radio technology have relatively coarse-grained performance indoors because their signals typically cannot be resolved accurately enough to produce quantities such as a time-of-arrival or angle-of-arrival. Instead, received signal strength indication (RSSI) or other metrics are used to estimate location via a technique called *fingerprinting*, but positioning errors of several metres often result because of indoor multipath fading. By contrast, ultra-wideband (UWB) systems can employ pulses of extremely short duration to achieve the much finer signal resolution. This resolution also aids in the identification of erroneous measurements due to multipath [63].

UWB positioning systems can measure time-of-arrival (TOA), time-difference-of-arrival (TDOA) or angle-of-arrival (AOA), or some combination of them. A time-of-arrival can be converted into a range estimate, but measuring time-of-arrival directly is problematic, because the synchronisation signal (typically conventional radio) and the UWB positioning pulse both travel at near the speed of light.* Pseudoranging (based on TDOA measurements) is more attractive for some deployments, since there is no need for precise synchronisation between the transmitting and receiving entities. A network of receivers can be precisely synchronised using stable clocks which are periodically corrected via a wired reference timing signal [187]. Producing AOA estimates requires a receiver equipped with an antenna array.

One UWB positioning system [62] estimates a tag's location using pseudorange data, with a stated accuracy of about 30 cm. Ubisense is a commercial UWB-based location system which performs measurement of both pseudorange and AOA (azimuth and elevation). The advantage of using AOAs as well as pseudoranges is that location can be determined with fewer sensors, compared to systems that use just pseudoranging. The reported accuracy (using their proprietary algorithms) in 3D is 15 cm.

Contributions: This chapter examines the results of two positioning algorithms for pseudoranging and angle data. The four primary contributions are as follows:

- A brief characterisation of the raw observations reported by two different Ubisense deployments (Section 5.3).
- Formulation of two algorithms (regression using a non-linear model and Kalman filtering Section 5.5) which fuse the heterogeneous observations (pseudorange, azimuth, elevation) of an UWB system.
- Characterisation of the static (Section 5.6) and dynamic (Section 5.7) tracking performance of the two algorithms.
- An evaluation of the algorithms on homogeneous data, showing that the algorithms are also appopriate for less sophisticated UWB positioning sensors, such as those which do not have tight synchronisation for pseudoranging, or are not equipped with arrays for sensing angle-of-arrival (Section 5.6.2).

*If a wired tether were used to connect the receiver and transmitter, direct range measurement would be possible after calibration of the timing offset due to the tether. Direct range measurement using round-trip time-of-flight measurements are possible for devices capable of both transmitting and receiving UWB positioning pulses.

5.2 Related work

Three approaches are generally used to calculate location using range, pseudorange, or AOA estimates. The first approach uses a simple geometric model to calculate intersection of circles (lateration), hyperbolas (hyperbolic localisation) or lines (angulation), depending on whether range, pseudorange, or AOA data is used. However, such simple algorithms typically do not take measurement error into account, and cannot make optimal use of redundant data (such as that gathered from a large number of receivers) which overspecifies the solution.

By contrast, a second approach is to use optimisation algorithms which are specifically designed to find a solution which minimises the total error between the collected data and the location estimate (i.e. the residual error). These algorithms traverse the solution space and compute expected measurements for each estimate of the solution. The *gradient* method employs derivatives to observe the rate at which an area of the solution space converges towards an optimum. Some examples in this category are the method of steepest descent, Newton's method, and the Levenberg-Marquardt method [159]. Such methods require *model equations* which are used to express the measured values (such as pseudorange, or AOA) in terms of the position being solved. The Bat system [189], and other ultrasonic positioning systems [55] apply such model-based optimisation algorithms to compute location solutions from ranging data.

Scott et al. utilise a non-linear system of equations to compute location based on pseudoranges (in this case TDOAs of acoustic signals) [93]. Fontana comments on the performance differences between the steepest decent search and Davidon-Fletcher-Powell algorithm for estimating a tag's location using UWB pseudorange data [62]. It is worth noting that these error minimisation algorithms do not make use of information obtained from prior location readings.

Algorithms that utilise a solution state (either current state, or current and past states) can be grouped under the third approach of *state-estimation algorithms*. State-estimation algorithms are used extensively in robot localisation. They operate by iteratively combining the previous estimate of the state (e.g. a position and orientation) with the observed measurements (range, AOA, etc.). Many state estimation algorithms exist [65], of which Kalman filtering [193] is most commonly used. The HiBall tracking system [194] employs a technique called single-constraint-at-a-time (SCAAT) tracking to model movement, and handles one observation at a time rather than obtaining multiple simultaneous measurements. The Constellation system [68] tracks a mobile unit consisting of a 3D inertial sensor and a number of ultrasonic sensors which report ranges that are fed into a SCAAT algorithm. The idea is to correct the positional drift in inertial tracking by incorporating ultrasonic range measurements in an extended Kalman filter. Smith et al. present a tracking algorithm using extended Kalman filtering of ultrasonic range data gathered from "Cricket" devices [173]. Smith et al. employ a combination of least squares minimisation, Kalman filtering and outlier rejection to predict the state of the Cricket device. While their approach is similar to our proposed Kalman filtering, we fuse heterogeneous data (pseudoranges and angles-of-arrival), and operate on UWB rather than ultrasound measurements.

Recent work by Renaudin and Kasser demonstrates the fusion of pseudoranges and AOAs from a Ubisense system together with inertial sensing data using an extended Kalman filter [164]. Curiously, their modelling equations are quite different to ours presented below (Eqns. 5.1–5.4) which we have verified through the analysis in this chapter. Moreover, the goal of Renaudin and Kasser is the *augmentation* of inertial tracking with UWB sensing. There is no literature which explores in detail the fusion of heterogeneous data (pseudorange and AOA) through regression of Kalman filtering for UWB position sensing.

5.3 Deployments and data collection

In all our experiments, we used hardware and software procured from Ubisense. In brief, Ubisense hardware is comprised of two entities: a tag† which emits UWB pulses when triggered by the system; and receivers ("Ubisensors") (figure 5.1(a)) which are typically fixed at upper extremes of the measurement volume. The receivers are networked via CAT5 cabling and it can be assumed that they are tightly synchronised (once the cable timing offsets are estimated during calibration). A workstation PC is connected to the same network, and runs the Ubisense Location Engine (LE), which can be used to configure and calibrate the system, and which produces location estimates from receiver measurements, using proprietary algorithms.

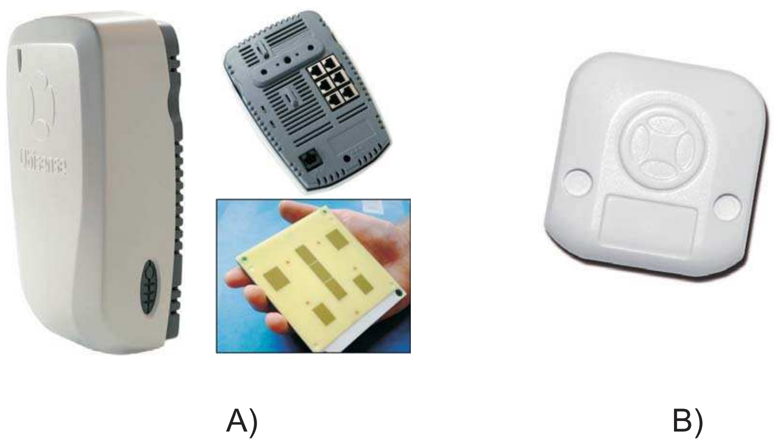
The calibration of the receivers' position and orientation are crucial in achieving accurate location estimates. The coordinates of a receiver's position can be estimated with sufficient accuracy in the order of few centimetres via manual methods, such as measuring the distance (using a tape measure or laser rangefinder) from the receiver centre to several known points in the environment. By comparison, accurate estimation of receiver orientation‡ (yaw and pitch) is more difficult without special equipment, and additionally there can be small misalignments between the plastic casing of the Ubisensor, and the plane of the UWB receiver array inside. Thus, calibration of receiver pitch and yaw is normally undertaken using a series of measurements from a tag and the measurement logging and orientation estimation process is automated by the Ubisense Location Engine. To calibrate the receiver pitch and yaw for the systems at our sites, we used two different modes of the automated process, as described below. The deployments we describe below were specifically made to differ in two respects: (i) granularity of the calibration i.e. calibrated using specialised equipment and system-specific knowledge vs. non-specialised and less accurate equipment (ii) test environment i.e. carefully chosen test points with favorable line-of-sight vs. randomly chosen test points with partially blocked/blocked line-of-sight conditions.

5.3.1 Low-overhead or real world deployment (Twente)

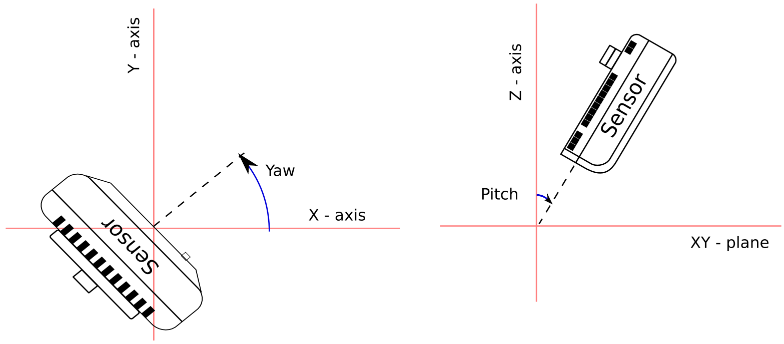
Six receivers were deployed covering an area of approximately 15×9 m (Figure 5.2b). Four of the receivers were deployed in an area which was relatively empty except for some desks along the corners of the room. The remaining two receivers were placed in the rooms along the other side of the corridor, in typical office spaces containing desks, metal shelving, and other furniture. We arbitrarily chose twenty-one test points across the measurement volume. Measurements were taken with tags placed at two different heights (75 cm and 151 cm above

† Ubisense offer both a "slim" and a "compact" tag. We chose to use the compact tag(figure 5.1(a))for the data collection. It is advertised that the tag emits pulses in an omnidirectional fashion, and it performed as such during our initial, informal experiments.

‡ Ubisense use the aeronautical terms yaw, pitch and roll to describe receiver orientation [183].



(a) Ubisensors (receivers equipped with antenna arrays and slot for provisioning time synchronisation) and Ubisense Compact tag (transmitter)



(b) Ubisense defined yaw and pitch: Yaw is defined as the positive anticlockwise and Pitch is defined negative anticlockwise

Figure 5.1: *Ubisense system entities and orientation.*

the floor). The positions of the sensors and the test points were surveyed using tape measures.

Automatic calibration. At Twente, we used the "automatic calibration" mode of the Ubisense LE. In automatic mode, a tag is placed at a position within the measurement volume. The height of the tag must be known and provided to the LE by the user, but the system then takes measurements to estimate the horizontal position of the tag, and the pitch and yaw of the receivers which can reliably detect the tag's pulses. The tag is then moved to another position and this process is repeated until the pitch and yaw of all receivers is estimated.

5.3.2 Carefully planned and calibrated or ideal deployment (Lancaster)

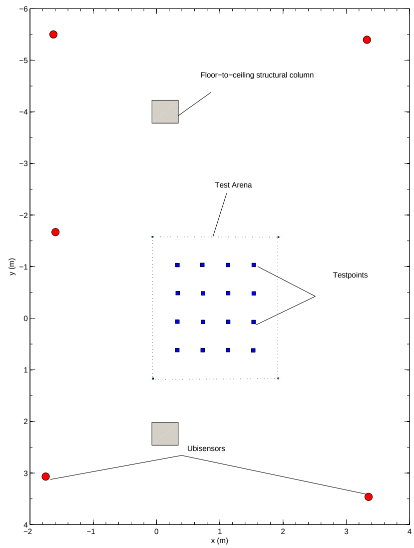
At another site, a smaller measurement area of 2.75×2 m was covered with five receivers. Readings were taken with a stationary compact tag placed at four heights (about 0, 7.5, 75, and 125 cm above the floor), at sixteen points across the measurement area (Figure 5.2a). The positions of the sensors and the test points were surveyed using a Leica Total Station.[§]

Dual calibration. At Lancaster, the "dual calibration" mode was used to estimate receiver pitch and yaw. In this mode, a tag is placed at a known position in the environment, and its 3D coordinates are provided to the LE. Measurements of the tag's pulses are then gathered from two receivers, and the receivers' pitch and yaw are estimated for one or both of the receivers (selectable by the user). The dual calibration has the potential to more accurately estimate pitch and yaw, since the estimation process does not depend upon the accuracy of the LE's result for the tag's horizontal position (as it does in the automatic calibration mode). Ubisense recommend that when performing calibration, the tag be placed as near as possible to the boresight of the receiver(s) being calibrated. At Lancaster, we ran the dual calibration for all five receivers at each of twelve known points (whose coordinates had been surveyed using the Totalstation) in or near the measurement volume. The twelve calibration points (which are different from the test points used in our evaluation) were each selected for their favourable line-of-sight to at least four of the five receiver units. To remove poor pitch and yaw estimates (typically due to poor line-of-sight or environmental reflections), we took the median of the twelve pitch and yaw values for each receiver as our final calibration.

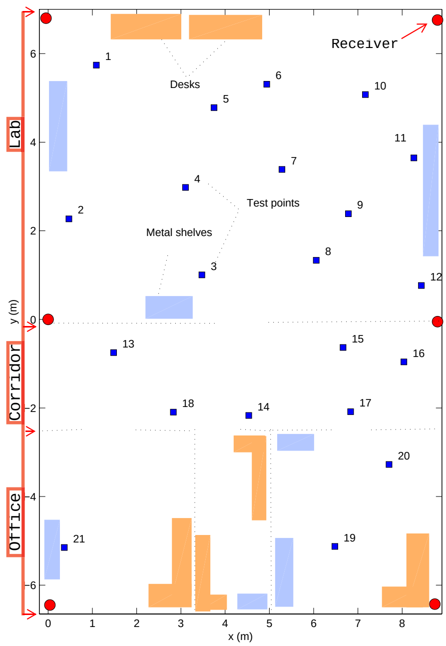
5.4 Characterisation of pseudoranges and angle-of-arrival

Figure 5.3 shows the distributions of the receivers' raw measurements for a static tag placed at the test points at each site (sixty-four locations at Lancaster, and forty-two locations at Twente). The accuracy of the Lancaster measurements is significantly better than that of the Twente measurements (Table 5.1). The Twente deployment covers a wider area, and not all receivers have line-of-sight to all the test points. In its measurement and location estimation process, the LE identifies receivers whose measurements were rejected for the purposes of computing a location; the rejected measurements typically correspond to receivers which had poor line-of-sight to the tag (and may therefore represent multipath readings). We plot the raw accuracy with these rejected measurements excluded. In total, the discarded measure-

[§]A Total Station is a professional surveying device which accurately measures the range, azimuth, and elevation between itself and a reflector, and which can thus be used to estimate the 3D coordinates of the surveyed point.



(a) Lancaster



(b) Twente

Figure 5.2: *Plan views of the deployment areas.*

ments represent 25% of the Twente dataset.

	Pseudorange	Azimuth	Elevation
Lancaster	6 cm	1.5°	9°
Twente	37 cm	22°	17°
Twente (excl. LE discards)	25 cm	6.8°	8°

Table 5.1: *Seventy-fifth percentile accuracy of the raw measurements of two Ubisense deployments.*

However, even with the LE-rejected measurements excluded, the data from the Twente deployment is significantly worse in accuracy when compared to the Lancaster data. In general, there are three sources of raw measurement error in any location system: (i) sensor inaccuracy; (ii) calibration inaccuracy (sensor position, pitch and yaw), and (iii) inaccuracies induced by environmental effects (attenuation and/or reflection of the UWB pulse). The Lancaster deployment was more carefully calibrated (Totalstation survey as opposed to hand measurements; and dual calibration of all receivers at twelve surveyed points, as opposed to automatic calibration at several arbitrary points). Moreover, the Lancaster deployment was specifically designed to accurately monitor the measurement volume; all five receivers have favourable line-of-sight to all sixty-four test points. By contrast, the Twente deployment has a much lower receiver density, covering a measurement volume containing office furniture and walls. For these reasons, we would argue that the inaccuracy of the Twente raw measurements is due to a poorer calibration (visible in the significant error offsets for some of the Twente receivers), and less favourable line-of-sight and multipath conditions caused by the environment.

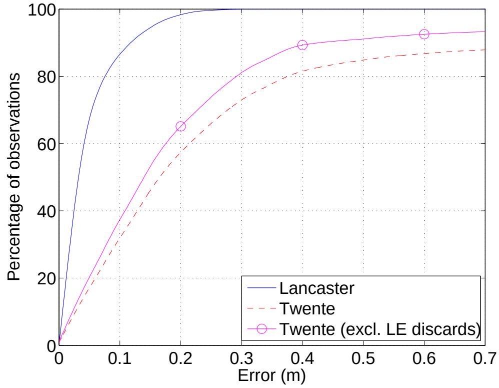
Thus, the two sites represent very different kinds of deployment. The Lancaster deployment has been carefully calibrated using specialised equipment and system-specific knowledge, and has been designed to monitor a small volume using a high sensor density. The Twente installation covers a much larger volume, the environment was left unmodified (leading to a higher degree of unfavourable pulse propagation for some tag locations), and sensor position and orientation have been calibrated based on fewer measurements taken with non-specialised, less accurate equipment.

5.5 Overview of Algorithms

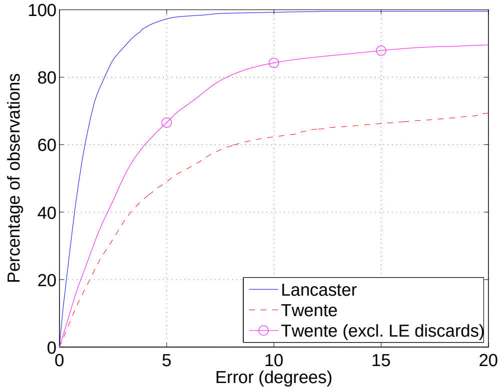
5.5.1 Non-linear regression

We formulate a non-linear regression algorithm [159, section 15.5] which utilises modelling equations. The regression is an iterative process which finds an estimate of the tag's location. The estimate can be seen as a "best fit" for the pseudoranges and/or angles-of-arrival and the surveyed receiver locations and orientations, since it minimises the sum of squares of the residual errors.

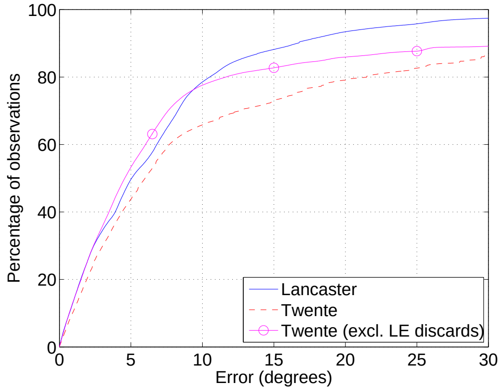
Pseudoranges. For a receiver i, the receiver's pseudorange estimate (based on the measured relative arrival time of the pulse emitted by the tag) $\tilde{d}_i$, the receiver's 3D location



(a) Pseudorange



(b) Azimuth



(c) Elevation

Figure 5.3: *Raw measurement error distributions.*

(x_i, y_i, z_i), and the tag's location (u, v, w) can be related as follows:

$$\tilde{d}_i = \sqrt{(u - x_i)^2 + (v - y_i)^2 + (w - z_i)^2} - d_c, \tag{5.1}$$

where d_c is the distance offset common to all receiver pseudorange measurements, and arises from the tag's unknown clock offset from the system. It is assumed that the receivers are very tightly synchronised, or their reported pseudoranges have been otherwise appropriately adjusted (using for example deployment-specific calibration information) for any time offset which may exist between receiver units.

Angles-of-arrival. The receiver-reported azimuth ϕ_i of a tag is related to the tag's position (u_s, v_s, w_s) in the receiver's frame of reference.

$$\text{Quads. I \& IV: } \phi_i = \arctan\left(\frac{v_s}{u_s}\right) \quad \text{Quads. II \& III: } \phi_i = \left(\pi + \arctan\left(\frac{v_s}{u_s}\right)\right) \tag{5.2}$$

Similarly, the elevation θ_i of a tag measured by a receiver is defined as

$$\theta_i = \arctan\left(\frac{w_s}{\sqrt{u_s^2 + v_s^2}}\right) \tag{5.3}$$

Note that a four-quadrant definition of the elevation θ_i is unnecessary, since the denominator of the arctangent operand is always positive.

To compute the tag location in the receiver's frame of reference, first the coordinates of the tag $(u_{rel}, v_{rel}, w_{rel})$ relative to the receiver are computed by subtracting the receiver coordinates from the coordinates of the tag in the global frame of reference, i.e. $(u-x_i, v-y_i, w-z_i)$. This effectively translates the origin of the global coordinate system to the location of the receiver. These tag-to-receiver relative coordinates must be transformed to the receiver's frame of reference. This is first a rotation about the Z axis by an amount corresponding to the receiver's yaw φ_i, followed by a rotation about the Y axis by an amount corresponding to the receiver's pitch ϑ_i. The rotation matrix R_{GS} can thus be defined as:

$$R_{GS} = \begin{bmatrix} cos(\vartheta_i)cos(-\varphi_i) & -cos(\vartheta_i)sin(-\varphi_i) & sin(\vartheta_i) \\ sin(-\varphi_i) & cos(-\varphi_i) & 0 \\ -sin(\vartheta_i)cos(-\varphi_i) & sin(\vartheta_i)sin(-\varphi_i) & cos(\vartheta_i) \end{bmatrix} \tag{5.4}$$

Note that Ubisense yaw is defined as positive anticlockwise (looking at the XY plane from the Z+ direction), and Ubisense pitch is defined as negative anticlockwise (looking at the XZ plane from the Y+ direction). The position of the tag (u_s, v_s, w_s) in sensor's frame of reference is then computed by multiplying the tag's coordinates relative to the receiver $(u_{rel}, v_{rel}, w_{rel})$ by R_{GS}.

Outlier rejection. The number of observations required (pseudoranges and/or angles-of-arrival) depends upon the particular tag-to-receiver geometry. In typical Ubisense deployments, to estimate both tag location and the distance offset $\hat{d}_c$ at minimum either two pseudoranges, one azimuth, and one elevation, *or* four pseudoranges are needed. To estimate only tag location, at least three angles-of-arrival (including at least one azimuth and one elevation) are needed. For locations estimated using more than the minimum number of observations, the standard error s of the estimate can be calculated from the residual errors e_i:

$$s = \sqrt{\frac{\sum_{i=1}^{I} e_i^2}{I - \mathrm{C}}}, \tag{5.5}$$

where I is the number of observations being used in the non-linear regression, and C is the minimum number of observations required. C is set to 3 if only tag location is being estimated, or to 4 if the distance offset $\hat{d}_c$ is also being estimated. The non-linear modelling process assumes the residual errors $\epsilon_{i=1\ldots I}$ are normal, independent, and have equal variance and zero mean.

With heterogeneous data such as that reported by Ubisense receivers (pseudorange, azimuth, and elevation), the computed residuals have different units. Thus, before applying Eqn. 5.5 to estimate the standard error, each residual should be scaled by a typical magnitude of error which might be expected for each type of data. To perform scaling of residuals, we utilised divisors of 30 cm, 4° and 3° for pseudorange, azimuth, and elevation, respectively. These were chosen to work with both the Twente and Lancaster datasets, rather than being specifically tailored to either one.

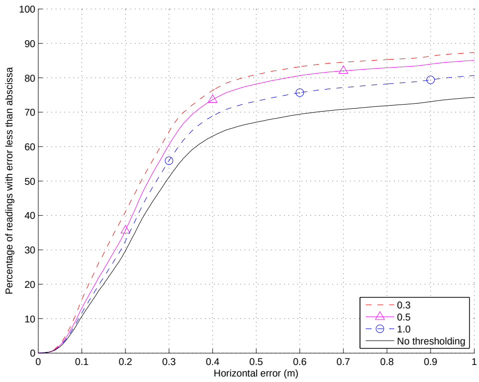
If the observations given to the regression do not corroborate one another, the result will typically have a high standard error estimate. Standardised residuals [71, ch. 4] can be used to identify the observations which agree the least with the solution. If the standard error estimate was greater than 0.30, our regression algorithm removes the observation with the highest standardised residual, and re-computes the solution. This process continues until the standard error of the estimate falls below 0.30, or the highest residual is less than 1.0.

Even after the outlying observations have been removed, the location accuracy can still be poor (Figure 5.4(a)). To improve accuracy, one tactic is to simply reject location readings which have a high standard error estimate. For the regression results presented in the remainder of this paper, we have omitted estimates with a standard error higher than 0.5 (except where otherwise noted). Assuming both pseudorange and angle-of-arrival data is used, this corresponds to about 20% of the readings taken at Twente (Figure 5.4(b)), and very few (less than 0.1%) of the static readings taken at Lancaster.

5.5.2 Extended Kalman filtering (EKF)

We formulate an Extended Kalman Filter (EKF) using a state vector $\hat{x}_k$ with seven variables, three position variables u, v, w, user clock offset d_c and three velocity variables v_u, v_v, v_w. After any discrete time step, which is approximately 108 ms[¶] the filter has an idea of its state and how confident it is in that state (prediction). The filter then corrects the predicted state based on the most recent measurements (pseudoranges and angles) and its internal state. Note that we use an Iterative Extended Kalman Filter (IEKF) [144], which is an extension to the

[¶]The tag emits a UWB pulse once every four Ubisense time slots. With our particular version of Ubisense hardware, each time slot corresponds to 27.029 ms.



(a) Horizontal error distribution

Std. error threshold	75% horiz. error (cm)	Fraction accepted (%)
0.3	38.4	64.2
0.5	42.1	80.4
1.0	57.0	91.2
None	106.4	100

(b) Trade-off of accepting regression solutions based on standard error

Figure 5.4: *Horizontal accuracy at Twente for different standard error thresholding levels.*

standard EKF, and is useful for reducing the errors that may occur due to large non-linearities in the system.

Initialisation. The filter is initialised with a posterior state estimate $\hat{x}_k^-$ and uncertainty P_k^-. Kalman filter estimates rely heavily on these initial estimates. We set the initial state estimates based on averaging the first twenty measurements through non-linear regression (as described above), discarding any with high standard error. Since we have set the tag at the highest update rate, this corresponds to a wait of a few seconds to start the Kalman filter. We chose to use a small but non-zero value for P_k^-, meaning that there is a little uncertainty in the defined initial state.

System Model and Measurement Model. We use a *constant-velocity model*, i.e. it is assumed the tag moves at constant speed between time steps. Thus, the new state estimate $\hat{x}_k$ will depend on the previous state estimate $\hat{x}_{k-1}$, constant velocity v_{xk} and a noise term w_k (as in, $\hat{x}_k = \hat{x}_{k-1} + v_{x_k} dt + w_k$). In order to predict the state using the measurements, we will have to describe how the measurements are related to the state. The *measurement model* $\hat{z}_k (= H\hat{x}_k + v_k)$ describes how measurements depend on the state estimates $\hat{x}_k$. H is the Jacobian matrix with partial derivatives of the measurement function with respect to the state $\hat{x}_k$. The measurement function here represents the pseudorange ($\tilde{d}_i$), azimuth (ϕ_i) and elevation (θ_i) as defined earlier in Sect. 5.5.1.

Prediction & Correction. The predicted error covariance (P_k^-) and the state estimate ($\hat{x}_k^-$) for a time-step is given by:

$$
\begin{aligned}
\hat{x}_k^- &= A\hat{x}_{k-1} + Bu_k \\
P_k^- &= AP_{k-1}A^T + Q.
\end{aligned}
\tag{5.6}
$$

Here, A is the Jacobian matrix and Q is the process noise covariance. The process noise covariance Q for a position-velocity model includes the process covariance in position and velocity and the process covariance in the position.

The filter computes the posterior state estimate by taking the prior state estimate and combining with the Kalman gain K_k times the difference between the actual measurement (pseudoranges and angles) and a measurement prediction ($\hat{z}_k = H\hat{x}_k + v_k$), called the *innovation* or residual r. If the innovation is zero, then the predicted state estimate exactly reflects the real measurement. But if there is a difference between the predicted and the observed measurement, then the prior state estimate needs to be updated. In Eq. 5.7, K_k determines to what extent the innovation should be used in the posterior state estimation. Based on the measurement noise R and the prior error covariance P_k^-, the gain can favour the innovations or the measurements more. The measurement noise R for pseudorange and angles (azimuth and elevation) is set to be 10 cm, 7° and 10° respectively.

$$
\begin{aligned}
K_k &= P_k^- H^T (HP_k^- H^T + R)^{-1} \\
\hat{x}_k^{i+1} &= \hat{x}_k^- + K_k(z_k - H(\hat{x}_k^- - \hat{x}_k^i)) \\
P_k &= (I - K_k H)P_{k-1}^-
\end{aligned}
\tag{5.7}
$$

In Eq. 5.7 the Jacobian matrix H is evaluated at the most recent intermediate state estimate $\hat{x}_k^i$ (difference between the IEKF and EKF). After a number of iterations or when the intermediate state estimate does not differ with more than a certain threshold from $\hat{x}_k^{i-1}$, the filter sets the posterior state estimate and estimates its posterior uncertainty. It is important to note that the IEKF computes the uncertainty in the state only after it finds the most accurate intermediate state estimate. Though the computations involved in IEKF are larger than the standard EKF, the state estimates will be better because of re-evaluation of the measurement function and the Jacobian.

Validation gating. It is imperative to employ some form of outlier rejection (also known as validation gating in Kalman literature) as part of the filter, since noisy measurements can cause the filter arrive at a bad state estimate. In most cases, innovation or residual r is used to identify outliers. An approach to eliminate outliers is based on $r^2S^{-1} > \gamma$, where S^{-1} is a scalar based on the state and γ is an empirically-chosen parameter [173]. Alternatively, the distribution of innovations can be used to detect innovations that are unlikely to occur. Recent work by Renaudin looks at the possibility of eliminating outlying measurements based on human body orientation and comparing the predicted state with the current measurements. Another interesting approach is to check the condition for *optimality* of the filter.

It is reported [134] that through the usage of statistical methods it is possible to check if the innovation sequence is *white* (a sufficient and necessary condition for testing optimality). If the filter shows sub-optimality, R and Q can be adjusted in order to make the filter optimal, thereby making the Kalman filter *adaptive*. We however, chose to use the simple strategy of using the innovation distribution (based on setting thresholds determined empirically) to identify outlying measurements.

5.6 Static Positioning

For the static positioning experiments at Lancaster, a tag was placed at the sixteen test points (Figure 5.2(a)), at four different heights. At each of these sixty-four locations, receiver readings were gathered for about one thousand tag pulses and about 1000 readings were gathered. At Twente, a tag was placed at the twenty-one test points (Figure 5.2(b)) at two different heights, for a total of forty-two locations, with about 1500 readings at each. For our experiments at both sites we set the tags to run at the highest update rate, emitting an UWB pulse once every four Ubisense time slots, which in our particular version of the Ubisense hardware is equivalent to about ten times per second. All receiver observations were fed into our algorithms, even those rejected by the Ubisense LE for its location estimation (see Figure 5.3).

5.6.1 Results from heterogeneous observations

Figure 5.5 shows the accuracy of the regression and Kalman filtering algorithms when they are provided with the heterogeneous measurements (pseudoranges and angles-of-arrival) produced by the receivers. Note that the vertical accuracy at Lancaster is more than thirty centimetres worse than the horizontal accuracy. This might be explained in two ways. First, the geometry of the Ubisense deployment (receivers are approximately co-planar) and the types of measurements they take (pseudorange, azimuth, and elevation) mean that there is more information about the tag's horizontal position solution (heavily contributed to by pseudorange and azimuth), and less about the vertical position of the tag (primarily affected by the elevation readings). Second, referring to Figure 5.3, one can see that the pseudoranging and azimuth accuracy tend to be better than the elevation accuracy. At Twente, the opposite is true: the vertical accuracy tends to be superior to the horizontal. However, about 30% of the Twente raw azimuth readings are off by more than 20°, whereas the elevation accuracy is comparable to that at Lancaster (Figure 5.3c).

On the Lancaster data, the Kalman filter performs slightly better than the regression. This is because the Kalman filter utilises its predicted estimate for the position of the tag, in addition to new receiver observations, to produce the new tag estimate. By contrast, the regression algorithm utilises only the receiver observations from the current time slot to compute a solution. Thus the Kalman filter can be slightly more accurate, especially for static situations where the location estimate is stable for consecutive observations. Note that for the Twente data, the Kalman filter has much worse horizontal error than the regression. However, this is because the regression rejects results with a high standard error estimate. Rejecting no readings (Figure 5.4), the regression's 75% horizontal accuracy of 106 cm is comparable to the Kalman filter's 108 cm.

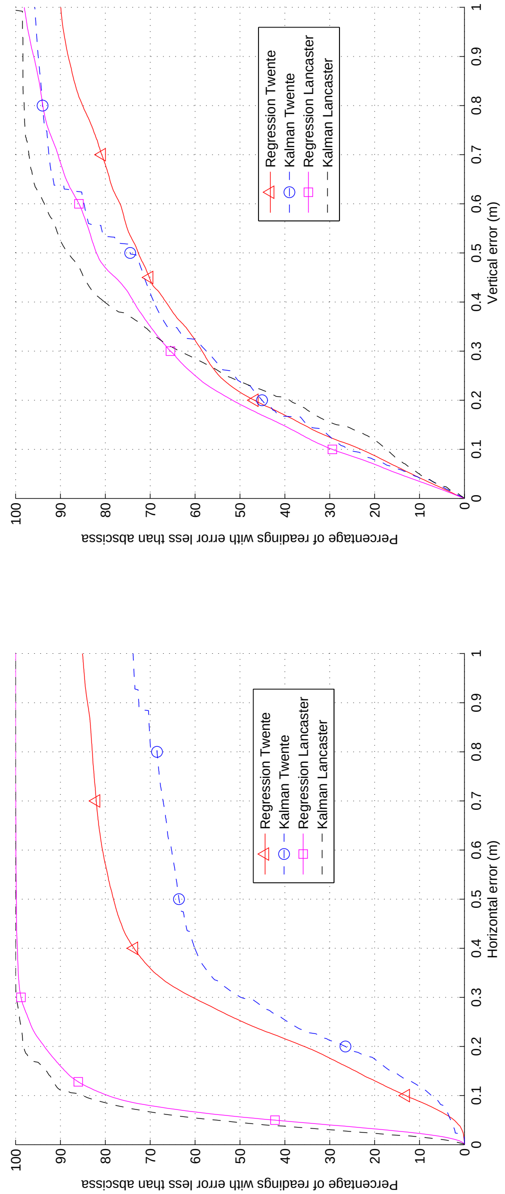


Figure 5.5: *Positioning static tags using heterogeneous measurements (pseudoranges and AoAs).*

		75% confidence level (cm)		90% confidence level (cm)		Estimates
Algorithm	Site	Horizontal	Vertical	Horizontal	Vertical	accepted (%)
Regression	Twente	42.07	54.74	155.59	101.04	80.42
Kalman	Twente	107.82	51.86	276.95	63.72	N/A
Regression	Lancaster	8.83	42.13	16.02	68.46	99.98
Kalman	Lancaster	7.51	37.35	11.14	52.61	N/A

Table 5.2: *Performance summary of static positioning using heterogeneous data.*

Effect of reduced receivers. Decreasing the required infrastructure can reduce installation and calibration cost. To show how well our algorithms support reduced infrastructure, we compute the tag locations based on the observations from subsets of receivers. As expected, the accuracy of both algorithms decreases as infrastructure density decreases. However, the Kalman filter degrades much more gracefully; note in particular the difference in the vertical accuracy results for Kalman and regression. Moreover, the regression rejects more readings as deployment density decreases, whereas the Kalman filter consistently supplies reliable estimates regardless of density.

	No. of	75% confidence level (cm)		90% confidence level (cm)		Estimates
	receivers	Horizontal	Vertical	Horizontal	Vertical	accepted (%)
Non-linear regression	All	8.83	42.13	16.02	68.46	99.98
	4	11.38	47.59	18.25	75.05	99.66
	3	15.02	55.82	22.85	86.29	95.07
	2	24.24	64.76	36.29	108.11	52.78
Kalman filtering	All	7.51	37.35	11.14	52.61	Not applicable
	4	9.10	37.38	13.69	53.02	
	3	11.38	37.60	17.07	51.93	
	2	14.45	35.74	17.26	53.87	

Table 5.3: *Reducing deployment density (heterogeneous data, Lancaster static measurements).*

5.6.2 Results from homogeneous observations

Both the algorithms we have presented are capable of computing location estimates using only pseudoranges or angles-of-arrival. It is interesting to consider their performance on homogeneous data in order to judge our algorithms' application for UWB position-sensing platforms having fewer capabilities than Ubisense. For example, receivers not equipped with array processing would not be capable of measuring AOA. Or, nodes in a distributed, wireless sensor network might only be able to measure angle-of-arrival, as the nanosecond-level node synchronisation required for pseudoranging is prohibitively difficult without a wired connection.

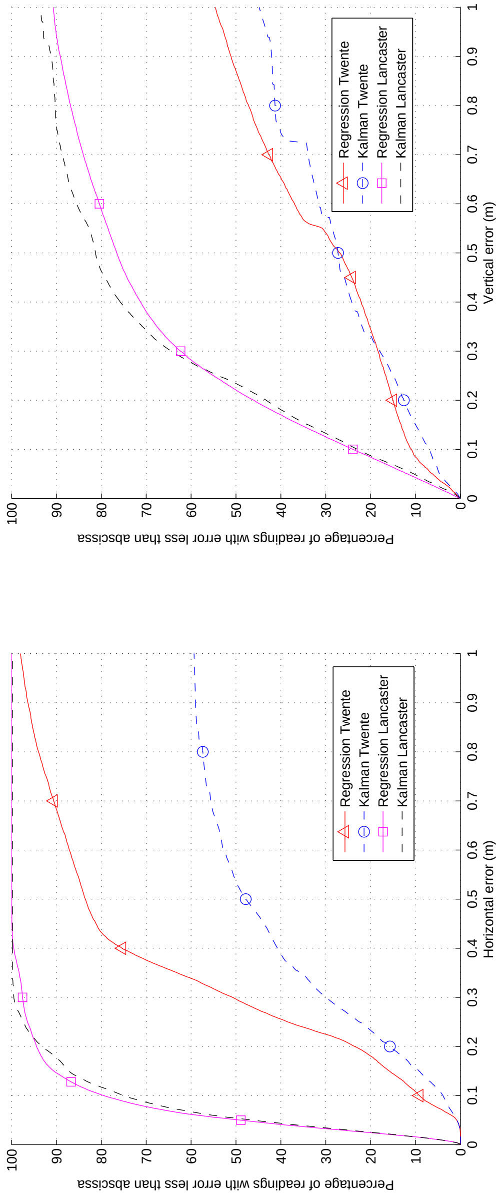


Figure 5.6: *Positioning static tags using pseudoranges only.*

Data type	Algorithm	Site	75% conf. level (cm)		90% conf. level (cm)		Estimates accepted (%)
			Horizontal	Vertical	Horizontal	Vertical	
Pseudo-ranges	Regression	Twente	39.80	141.28	68.30	233.93	18.48
	Kalman	Twente	330.62	491.51	1983.08	1229.96	N/A
	Regression	Lancaster	8.85	46.48	14.66	94.44	84.14
	Kalman	Lancaster	10.03	39.17	17.75	75.80	N/A
AoAs	Regression	Twente	190.55	64.10	435.11	103.02	63.93
	Kalman	Twente	452.10	59.62	678.16	119.24	N/A
	Regression	Lancaster	19.56	38.95	30.55	70.40	98.1
	Kalman	Lancaster	17.18	37.15	26.89	54.92	N/A

Table 5.4: *Performance summary of static positioning using homogeneous data.*

Pseudoranges only. Figure 5.6 shows the accuracy of the two algorithms operating on pseudorange data only, at both sites. As in the heterogeneous case, the regression performs significantly better than the Kalman filter on the Twente data; however note the ratio of readings rejected by the regression—over four-fifths for the Twente dataset (Table 5.4). Even for the Lancaster dataset, the regression rejects over 15% of readings. On the Lancaster data, the Kalman filter's accuracy is comparable, and it provides constant updates (no readings are rejected). Note that in general the vertical accuracy is inferior compared to the heterogeneous case. This is because the vertical solution is less constrained when only pseudoranges are used (no elevation data is present). The 3D accuracy of the Twente results are especially poor, since the accuracy of the underlying pseudoranges is much lower than in the Lancaster deployment (37 cm compared to 6 cm, at the 75% confidence level).

Angles-of-arrival only. Using only angles-of-arrival, the two algorithms perform comparably (Figure 5.7 and Table 5.4). The accuracy for the Lancaster deployment is quite favourable (horizontal and vertical 75% confidence better than 20 and 40 cm, respectively). This is because for the majority of the Lancaster readings, a reliable azimuth and elevation were reported by all five receivers. The Twente AOA-only results are quite the opposite of the Twente pseudorange-only results. While the Twente vertical accuracy is about as good as the heterogeneous case (Figure 5.5), around half of the readings exhibited a horizontal accuracy worse than one metre. Inspecting the per-receiver distributions reveals that three out of the six Twente receivers returned very poor azimuth estimates about 25% of the time. As mentioned above, this is most likely due to environmental effects and an inaccurate calibration. Thus, without the pseudorange contribution to the solution, the seventy-fifth percentile horizontal accuracy falls from several tens of centimetres in the heterogeneous case, to several metres in the AOA-only case.

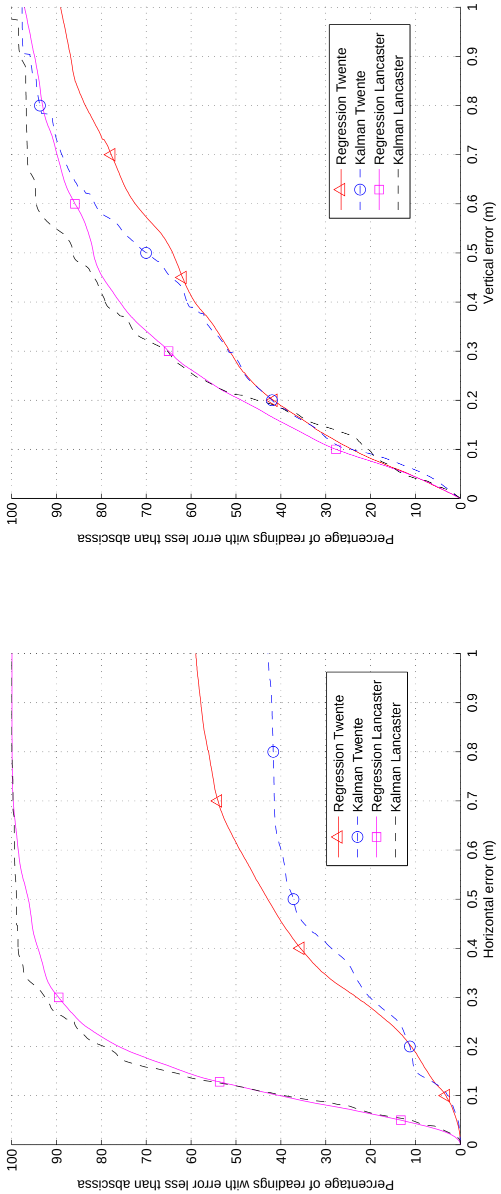


Figure 5.7: *Positioning static tags using angles-of-arrival (azimuth and elevation).*

5.7 Dynamic Tracking

At Lancaster, dynamic data traces were taken in the test arena. Data was recorded (Figure 5.8) in traces lasting approximately seven minutes each, resulting in about 2500 pulse readings per trace. We gathered two types of dynamic data (Figure 5.9): "robot" data (four traces) was generated as a Lego Mindstorms robot roamed the arena (velocity 0.16 m/s); and "walking" data (three traces) was recorded as a person pulled the robot around in the arena using a tether (peak velocity of about 1 m/s). The groundtruth positions of the tag in the dynamic experiments were recorded using computer vision–based localisation. Two cameras equipped with fisheye lenses were placed above the measurement area, a fiducial marker was rigidly attached to the top of the tag, and reacTIVision software was used to perform accurate localisation in real time. Since there are slight differences in the timestamps between the logged camera estimates and the ubisense estimates, we use a weighted-average method to interpolate the camera data for the corresponding ubisense timestamp. Where there were large gaps in the camera timestamps, the interpolated values may not be valid, hence interpolation was done only if the surrounding data points are close in time (shorter than 300 ms). For gaps greater than this, the Ubisense readings are discarded, to allow accurate comparison. For most traces, this method resulted in about 10% of receiver observations being dropped.

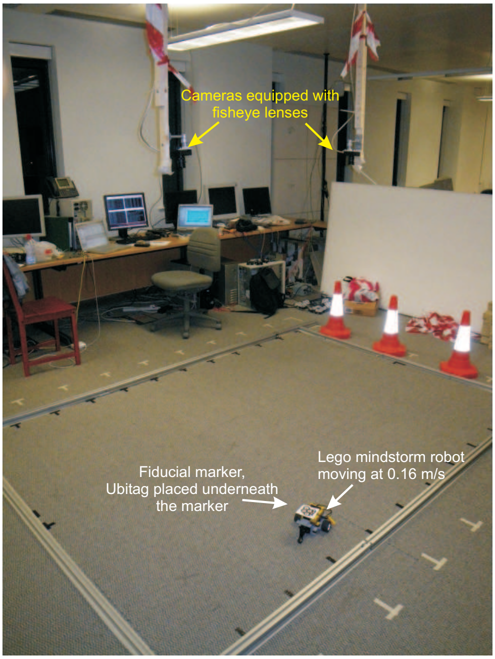


Figure 5.8: *Test arena, Lancaster dynamic experiment.*

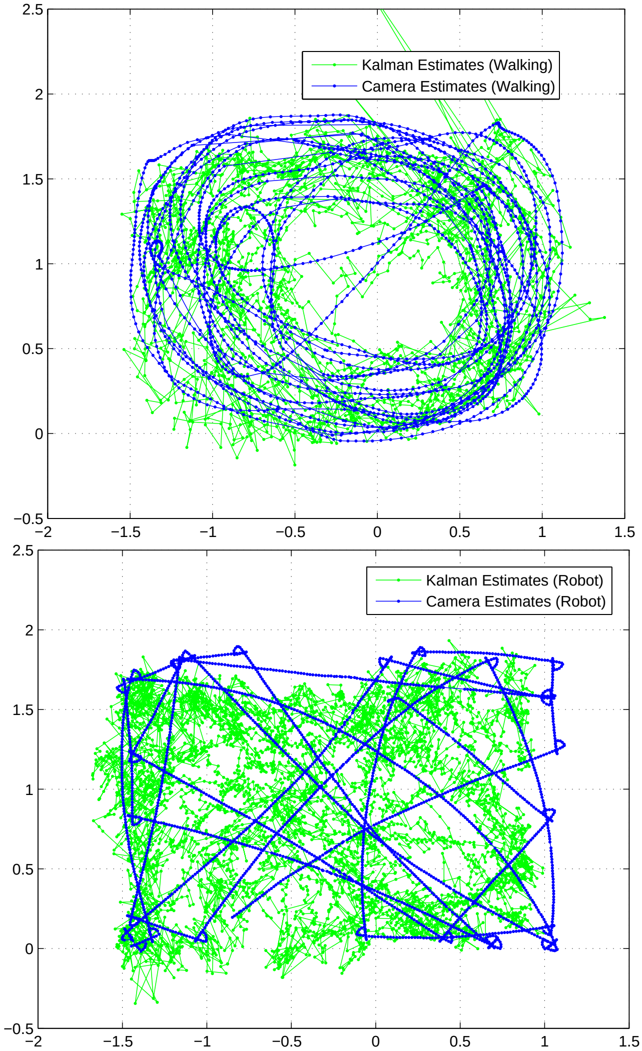


Figure 5.9: *Sample location traces: "Walking" (top) and "robot" (bottom), Lancaster dynamic experiment.*

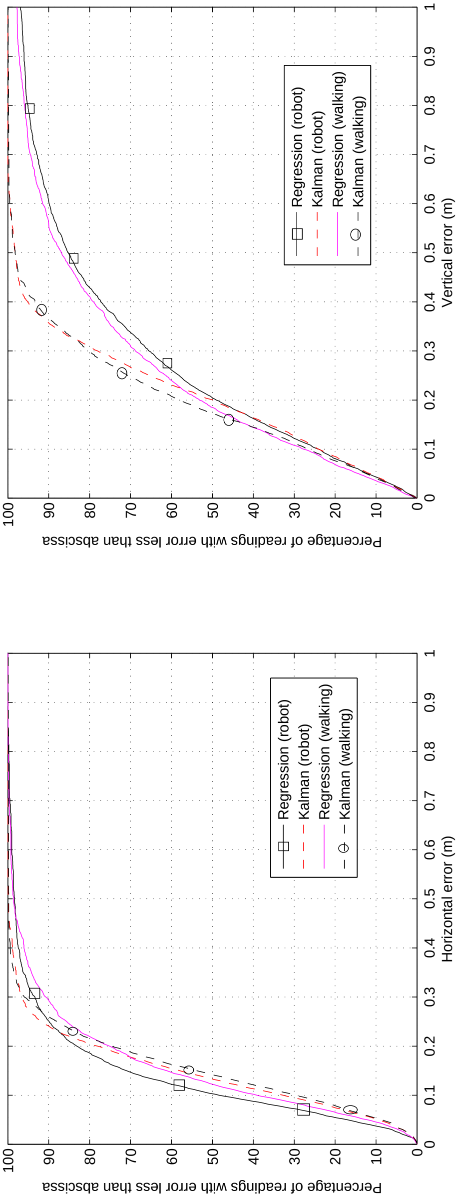


Figure 5.10: *Dynamic tracking of tags using heterogeneous data (Lancaster).*

The location accuracy for the "robot" and "walking" traces is roughly equivalent for each algorithm, despite the large difference in the speed of the tag (Figure 5.10). This is likely due to the sufficient update rate of Ubisense readings (about 10 Hz) for both types of trace. Since the accuracy does not change noticeably between the two speeds, for the homogeneous dynamic case (Figure 5.11 and Table 5.5), we consider only the three "walking" traces. The "walking" homogeneous results are comparable to the estimates for static data taken at similar height. For example, the regression operating on pseudoranges only yields a 75% horizontal dynamic accuracy of 20 cm, which is exactly the same accuracy as for the static readings taken at a height of 7.5 cm.

Data type	Algorithm	75% conf. level (cm)		90% conf. level (cm)		Estimates accepted (%)
		Horizontal	Vertical	Horizontal	Vertical	
Pseudoranges and AoAs	Regression	18.31	35.31	27.46	55.43	99.66
	Kalman	21.02	26.07	31.42	36.12	N/A
Pseudoranges only	Regression	20.29	90.63	26.86	167.24	61.86
	Kalman	33.30	46.27	55.84	53.90	N/A
AoAs only	Regression	32.43	39.58	48.53	61.01	94.83
	Kalman	48.69	15.87	62.87	22.16	N/A

Table 5.5: *Dynamic tracking performance summary (Lancaster). All values shown pertain to the location results of the three "walking" data traces.*

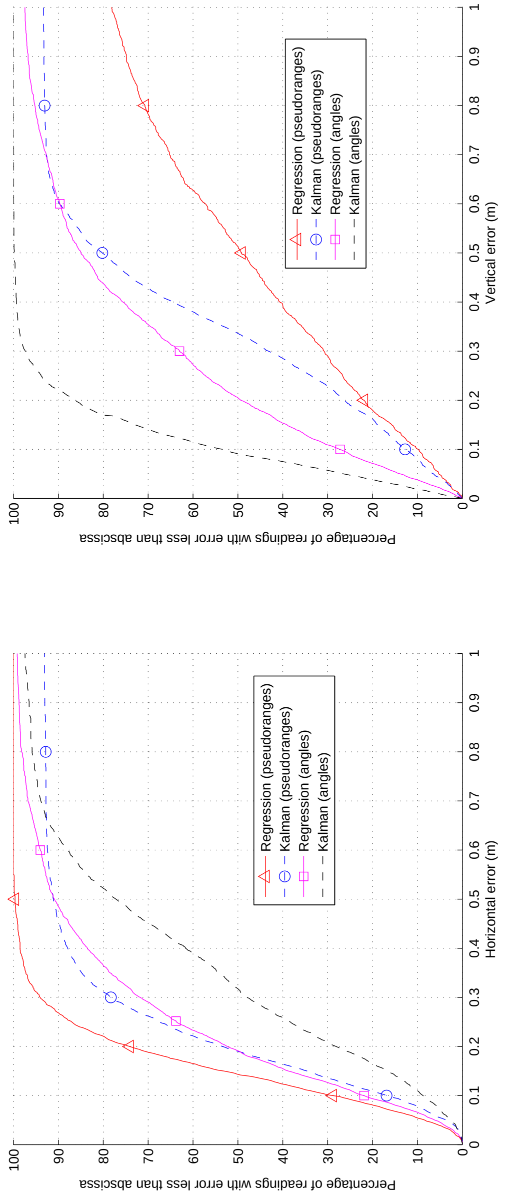
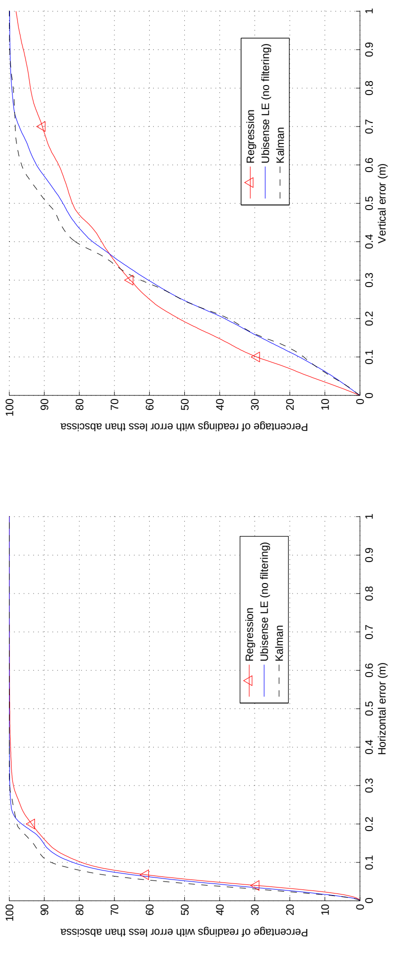


Figure 5.11: *"Walking" tracking accuracy using homogeneous data (Lancaster).*

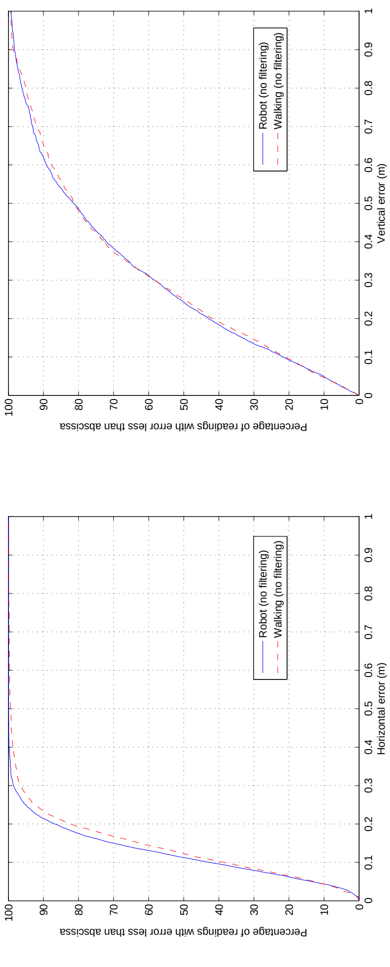
5.8 Ubisense Location Engine Estimates

Although the goal of this chapter is to characterise our algorithms' performance with different deployments and types (pseudorange and angle-of-arrival) of UWB positioning data, we provide here the estimates produced by the Ubisense Location Engine for purposes of comparison. When the data was logged at Twente, the Ubisense LE filtering parameters had not been set appropriately—for all experiments at Twente, the LE was configured to "fixed height information filtering," which requires an approximate height to be set by the user. Our algorithms assume no such special knowledge, so we present only results gathered using the Lancaster deployment.

Figure 5.12(a) shows the static tag positioning accuracy of the LE when configured to "default no filtering." The regression and Kalman algorithm results for the same data are also shown. Like our regression algorithm, the LE produces a measure of standard error. But, because we do not know how the LE standard error is calculated, for this comparison we have not rejected any estimates from the LE or the regression results. With no filtering, the Ubisense LE performs quite comparably to our proposed algorithms for static tags. Likewise, comparing Figure 5.12(b) to Figure 5.10, the dynamic accuracy of the LE is similar to our results if "no filtering" is set.



(a) Static performance comparison of LE results with the presented algorithms (Regression and Kalman Filtering)



(b) Dynamic performance of Ubisense LE

Figure 5.12: *Ubisense Location Engine accuracy (Lancaster deployment).*

5.9 Conclusions and Future work

This chapter has presented a non-linear regression and a Kalman filtering algorithm designed specifically to process the heterogeneous data which UWB positioning systems are capable of producing. Both algorithms fuse the different types of raw data (pseudorange, azimuth, and elevation) effectively. For reliably accurate raw data (as produced by the Lancaster deployment), the algorithms exhibit similar performance, and we would select the Kalman filter since it provides a more consistent (if at times slightly less accurate) stream of location estimates. Supplied with reliable readings, the Kalman filter performs better than regression as deployment density decreases. For deployments with poor calibration and/or less reliable, “noisy” readings (as in the Twente data), we would select the non-linear regression algorithm for its accuracy, despite the high ratios (20–80%) of rejected readings.

We have shown that the algorithms can work well on homogeneous data (pseudoranges or AOAs), despite the reduction of the information contributing to the location solution. Under certain configurations, noisy, homogeneous data can be pathological, as in the Twente pseudorange-only vertical accuracy, or the Twente AOA-only horizontal accuracy. When working on homogeneous data with reliable accuracy, the algorithms continue to produce good location estimates, as seen in the Lancaster results.

One may note that the accuracy of the Lancaster horizontal accuracy (typically 8–10 cm with 75% confidence) is *worse* than the accuracy of the raw pseudoranges (6 cm). Higher accuracies could perhaps be achieved by lowering the pseudorange variance estimate (30 cm in our algorithms), which would have the effect of weighting the pseudoranges more heavily. However, we wanted to design our algorithms to work on both datasets without modification. Of course, to squeeze the best performance out of any algorithm, its parameters should always be set according to the data expected from the particular deployment.

In many of our results, the vertical location accuracy was worse than the horizontal location accuracy. As noted previously, this is very much a function of the geometry of the receivers in our deployments. One can envision receiver deployment strategies which correct the bias, such as fixing some receivers high on the wall (as in our test deployments), and some receivers to the ceiling with their boresight facing the floor. The ceiling-mounted receivers' reported AOAs would contribute more heavily to the horizontal estimate, and the pseudoranges (in combination with pseudoranges received from wall-mounted receivers) would contribute much more to the vertical estimate.

We have compared the results of our algorithms with that of the estimates from Ubisense LE that are based on proprietary algorithm. While the results are quite comparable, reproducing similar results brought us closer to understanding how out-of-the-box commercial systems work and how effective does the system operate on different environments and calibration strategies. Our analysis of the raw measurement errors and its relation to different deployment methods highlights the requirement of expert knowledge for setting up a location system.

As future work, we plan to explore adaptive Kalman filters. This is particularly interesting as Kalman performance is heavily influenced by the choice of the model parameters, and making the filter adaptive can automatically tune the model parameters based on the current measurement. Usage of other probabilistic algorithms, for example particle filtering, may also be a fruitful area for investigation.

The use of a Kalman filter requires a state-space model for the dynamics of the process to be estimated, the tag's motion in our case. Although there are several models available to describe the dynamics of the moving user [41], we use a constant-velocity model. It would be interesting to explore different models and analyse their tradeoff's. It would certainly be an interesting step to evaluate the sensitivity of a particular algorithm to changes in the dynamic model parameters.

Another avenue of research is to extend the presented algorithms to address *online auto-calibration* of reference nodes. Called simultaneous localisation and mapping (SLAM) in robotics, this can also be used for calibration-sensitive infrastructure to bring us a step closer in realising easily deployable yet accurate positioning systems.

CHAPTER VI *

Ultrasound-aided pedestrian dead reckoning for tracking and navigation

Ad hoc solutions for tracking and providing navigation support to emergency response teams are an important and safety-critical challenges. The solutions based on inertial sensing systems are promising, but are subject to drift. Based on a brief characterisation of the errors encountered in inertial-based dead reckoning estimates, we propose navigation and tracking solutions based on a combination of foot-mounted inertial sensors and ultrasound beacons. The inherent drift of dead reckoning is addressed by deploying ultrasound landmarks. Simulation results show that satisfactory guidance performance is achieved by the proposed approach. However, smoother data is required by the guidance system to provide better experience for users. To account for this, we formulate two tracking algorithms that are based on Kalman filtering – (i) using only ultrasound data and (ii) using a combination of ultrasound and inertial data. We perform evaluation of both the tracking algorithms for data collected from real deployments for different trail topologies and show how filtering the raw data produces smoother and robust estimates.

*Part of this chapter is published with the title, *Ultrasound-aided pedestrian dead reckoning for indoor navigation*, In the Proceedings of Mobile Entity Location and Tracking in GPS-less environments (MELT), co-located with Mobicom, San Francisco, USA, September 2008 [60].

6.1 Introduction

In this chapter we focus on a firefighting scenario and the use of a sensor network to directly support firefighters in their rescue mission. Inertial navigation or pedestrian dead reckoning (PDR) has been applied to tracking and navigation of first responders with promising results. However, the position error in a purely inertial system increases with time and requires correction from external sources. A common practice is to periodically use GPS to correct position estimates [165], but for most indoor scenarios, GPS is unavailable.

We address the problem of positional drift by having the responders themselves deploy landmarks as they progress into an unknown environment. We specifically use ultrasound nodes. The *breadcrumb* trail (as shown in Figure 6.1) can be used to assist the PDR in guiding the responders back to their starting point, or guiding other responders towards a victim or an alternative exit.

Localisation is particularly critical in sensor networks conceived as an ad hoc location system for mobile agents (users, robots), as this involves tracking of the mobile node location relative to the location of the deployed nodes. Consequently, localisation has received considerable attention over the last years, both in WSN research [118] and in mobile and ubiquitous computing [90]. However, state of the art algorithms for WSN have two key limitations prohibiting their use for localisation in sensor trails: (i) most algorithms proposed for WSN require multiple beacon nodes with known position, whereas trails are developed from a single reference beacon and (ii) algorithms generally assume a scattered node distribution for multilateration, in contrast to a more linear topology along trails.

We use a combination of methods, namely ultrasound and inertial tracking (as shown in Figure 6.1), in order to tackle the problem of tracking in sensor trails. Such a system can be extended as an ad hoc location system for tracking and guiding mobile users. Neither method is sufficient for the task independently. For instance, ultrasound measurements have limited precision (outlier measurements due to multi-path effects and noise) and reliability (signal loss between neighbouring nodes due to communication or line-of-sight problems). Along trails, these limitations are critical due to scarcer connectivity. Inertial tracking has been shown to give good results for following short trails but is prone to large drift with increasing distance, and to scaling errors caused by disruptive motion (e.g. tripping, sidestepping, and sharp turns).

Our general tracking and navigation problem can be split into the following sub-problems: (i) locating the static nodes (deployed landmarks), (ii) tracking a pedestrian by fusing multimodal data and (iii) guiding the user to an appropriate destination or exit. In this work we focus on tracking and guiding the user using a combination of ultrasound and inertial measurements.

Contributions: The three key contributions are as follows:

- A brief characterisation of the errors encountered in an inertial-based pedestrian dead reckoning solution. We evaluate experimentally the performance of our dead reckoning system for data gathered from different environments and for different trail topologies. The characterisation of pedestrian dead reckoning (PDR) errors facilitates the need for a complementary sensing technology to correct for the drift in the inertial estimates.

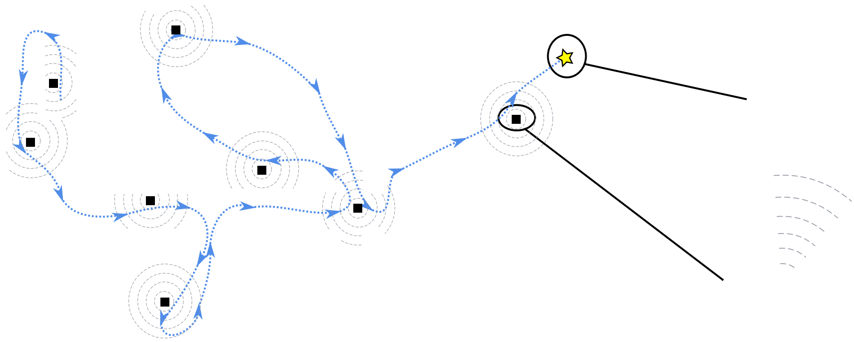


Figure 6.1: *Breadcrumb trails, with deployed ultrasound nodes shown as black squares and boots equipped with equivalent ultrasound node and inertial sensor.*

- Simulation of a guidance system to show the feasibility that a combination of inertial and ultrasound sensors would be effective for guiding the user along the trail of deployed nodes.
- Formulation of two tracking algorithms based on Kalman filtering –(i) using ultrasound measurements only and (ii) fusing inertial and ultrasound measurements. Characterisation of the dynamic tracking performance of the two algorithms based on traces of data gathered from real deployments.

6.2 Related work on tracking and guiding technologies

In this section we review some of the related work on tracking and guiding technologies that were developed specifically for search and rescue operations. Emergency response is an area where distributed sensing and localisation not only provide extra services to the users but are intended to save lives. Different sensing technologies have been used in literature to solve localisation and tracking problems in search and rescue missions. The Fire project [177] has developed SmokeNet, a wireless network of smoke detectors that are pre-deployed in the buildings. The firefighter wears a Mica2 Mote [8], a small wearable computer and equipped with a head mounted display. When a firefighter node enters a SmokeNet enabled building, the SmokeNet will identify this node as a firefighter and route messages to the node pertaining to the firefighter's location, the location of other firefighters, location of the fire, etc. Additionally the sensors will also monitor firefighters vital information such as heart rate and send the information to the other firefighters. The Flashlight by Peterson and Rus [155] guides a person through a sensor network avoiding danger zones by providing tactile feedback when they are facing the right direction.

The indoor positioning system [73] developed by Thales works similar to GPS but it is operational indoors: firetrucks parked around a building act as "satellites" that use UWB

RF signals to locate firefighters inside a building by means of time of arrival measurements. Although this system might perform well for lightweight residential buildings, UWB may not penetrate larger structures that extend underground for instance. For this reason we choose to deploy a physical chain of sensors that can create a link to the outside both for navigation and communication purposes.

Dead reckoning has the distinct advantage of providing autonomous positioning capabilities and is thus particularly attractive for indoor search and rescue operations. However, positions provided by this method will unavoidably drift over time due to errors in measurements being integrated [66]. The drift can be reduced by using shoe-mounted inertial sensors and resetting the velocity to zero at each footfall [149] and by combining the inertial measurements with data from an electronic compass through a Kalman filter in order to avoid drift in heading [67]. It has been shown that disruptive motion such as side-stepping, back-stepping, and tight turns which are typical in search and rescue scenarios produce scaling errors and cause the travelled distance to be over or under estimated. Thus, the estimated position drifts even more than during normal walking. Despite these limitations, dead reckoning is the only completely self-contained location technique that requires no prior knowledge of the environment. This is why we, and others attempt to address these limitations by combining dead reckoning with other complementary technologies.

In most cases it is essential to correct position and heading with data from external sources. GPS is one possibility but only for outdoor navigation with short periods of GPS outage [165]. Another possibility is to pre-deploy RFID tags at known locations and use these to correct positions [201]. Indoor location systems such as Ubisense have also been used in combination with PDR [86]. However there is no guarantee that a building will be equipped with any particular location infrastructure.

The navigation system developed by Renaudin et al. combines PDR with map matching in order to prevent drift [166]. Inertial measurement units (IMUs) on the chest and legs are used to measure movement and posture. The first team to enter the building deploy an RFID tag on each door frame as they pass through. The position computed by the inertial navigation system (INS) can then be corrected according to a database of the coordinates and directions of all doors in the building. The subsequent team members are equipped with an RFID reader and can therefore determine their positions as they scan each tag. This is an attractive solution since it is entirely ad hoc. Nevertheless it requires floorplans of the building and will fail in areas with few doors such as open plan offices or airport terminals.

We believe that RF-based sensors are not suited to indoor navigation because they do not account for walls. Ultrasound propagation on the other hand is inherently limited by walls and doors thus guaranteeing room-scale granularity or better. In our system we will use ultrasound nodes from the Relate project [82] as landmarks to correct for the drift in PDR. Ultrasound has also been used in several other location systems [160, 188] as mentioned in Chapter 2.

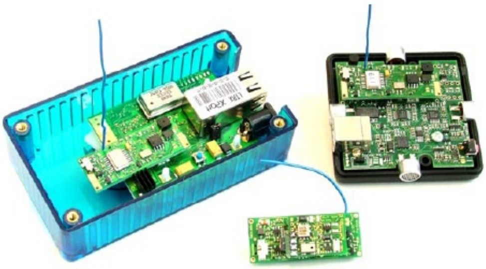
6.3 Preliminaries

We are basing our work on sensors that are available either as a research prototype or as a commercial product. Specifically, the ultrasound nodes that we use for the work presented here is developed as part of an European project called Relate [21]. The inertial sensing unit is from Xsens Technologies [199]. This section outlines the basics of the devices that are used for the work presented in this chapter.

Ultrasound Sensor

(a) Ultrasound node used for deployment as landmarks.

(b) Boots fitted with the flying leads



(c) Particle XBridge with ultrasound node and particle communication board.

Figure 6.2: *Relate ultrasound node as – landmarks, attached to the boots and particle communication board [21].*

The ultrasound nodes we use are known as "bricks" (shown on the right in Figure 6.2c).The sensor board consists of four 40 KHz narrow band ultrasound transducers, a temperature sensor and a power supply. The core processing and communication is handled by a Particle Computer with a PIC18 micro controller and 868 MHz RF chip. Communication primitives are provided by the AwareCon stack [179]. The ultrasound transducers act both as receivers and transmitters. When the brick is in the receiving mode, it uses data from transducers on which they detect ultrasonic pulses of sufficient strength and measure peak signal values and the TOF of the ultrasonic pulses sent by the transmitting device. The smallest TOF is then used to estimate the range. The AOA estimate is derived using the known orientation of the transducers on the brick and calculated based on the relative spread of peak signal values measured across these transducers. More details on the range and bearing estimation algorithms from these ultrasonic devices have been previously published [82, 196]. Flash memory (512K) on the nodes was available as non-volatile storage. This provided a big buffer capacity until data was read out. For maximum flexibility, instead of using a targeted peer-to-peer based scheme, all measurements were triggered externally by a data collection console and stored locally on the nodes for later retrieval. Data stored was automatically annotated using a local timestamp with a millisecond clock resolution.

Inertial measurement unit

The MTx [199] inertial measurement unit comprises of a tri-axis accelerometer, gyroscope and magnetometer. The on-board processor computes drift-free 3D orientation. In order to convert the MTx measurements into meaningful positions, the raw accelerations are rotated from the sensor coordinate system into the world coordinate system using the rotation matrix computed by the MTx as shown in Figure 6.3. More details on the characteristics of this device are available [199]. We detail the algorithm we use to perform dead reckoning based on the raw measurements (taken using accelerometers and gyroscopes), rotations computed by Xsens in Section 6.5.1.

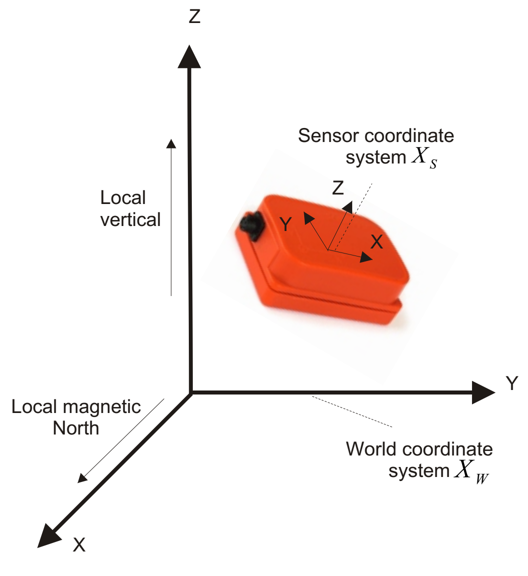


Figure 6.3: **MTx** *IMU from Xsens –Transformation from sensor to world coordinates via the direction cosine matrix:* $\mathbf{x_W} = R_{GS}\mathbf{x_S}$*[199].*

6.4 Characterising ultrasound range measurements

In this section we briefly characterise the raw range measurements of the ultrasound nodes used for the work presented in this chapter.

Figure 6.4: *Placement of ultrasound nodes used for raw measurement characterisation. Sender node placed at successive points for a few minutes, while the remaining nodes were made to listen for the ultrasonic transmissions from the sender.*

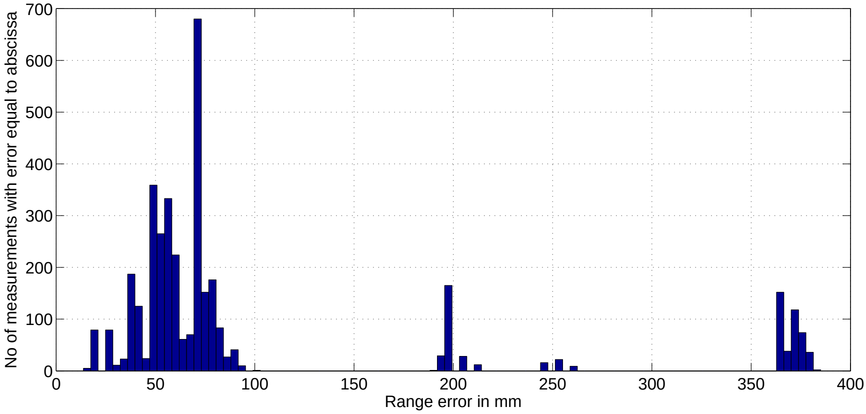


Figure 6.5: *Histogram of a typical set of range measurements gathered between five ultrasound nodes placed relatively close to each other at static locations. Each node takes turn to send out ultrasonic pulses for a few minutes which are received by the nodes in range.*

We placed five ultrasound nodes that are relatively close to each other as shown in Figure 6.4 with each node having average of three neighbors i.e. an average connectivity of three. Each time, one of the nodes acts as the sender while the others in range will receive the ultrasonic pulse emitted by the sender and calculate the distance/angles as reported [82, 196]. In total, approximately 4000 ultrasonic pulses were reported by all the receiving nodes. The error shown is computed as "measured value minus real value", so that negative distance errors indicate under-ranging, and positive distance errors indicate over-ranging. From Figure 6.5, we observe that the distances are typically over-estimated. Range measurements between four pairs of nodes mostly had errors less than 10 cm, (corresponding to the main gaussian block in Figure 6.5). In some cases error goes higher up to 40 cm. As with any ultrasonic ranging device, limited line of sight conditions cause performance degradation. When the line-of-sight between two devices is fully or partially blocked, several factors can contribute to the measurement error. Firstly, the tendency of ultrasonic waves to bend around obstructions can slightly lengthen the measured TOF, reduce the received signal strength, and cause the received pulse shape to vary from the expected shape of a direct-path pulse. Secondly, the receiver is more likely to identify multipath signals (i.e. reflections) as the valid ranging pulse. The brief characterisation of range errors reported here are in accordance with the characterisation of the nodes range and angle errors as reported by Hazas et al. [82].

6.5 Characterising Pedestrian Dead Reckoning

Dead reckoning is a self-contained navigation technique in which measurements — typically from inertial sensors — are used to track the position and orientation of an object given an initial position, orientation and velocity. No infrastructure is required but the position error will increase over time due to noise. In this section we describe how we convert the MTx's raw measurements (taken using accelerometers and gyroscopes) to dead reckoning estimates.

6.5.1 PDR algorithm

Our pedestrian dead reckoning algorithm is similar to other work [34, 67] which uses shoe-mounted inertial measurement units (IMUs) and applying periodic zero velocity updates (ZUPTs). In order to convert the MTx measurements into meaningful positions, the raw accelerations are rotated from the sensor coordinate system into the world coordinate system using the rotation matrix computed by the MTx (as shown in Figure 6.3). The accelerations are then double integrated to yield position estimates. In order to reduce the position error (which increases quadratically with time) we reset the integrated velocities to zero at each step resulting in linear error with distance covered.

Two phases in walking are identified: the stance phase, when the foot is in contact with the ground, and the swing phase. During the stance phase the velocity is reset and kept at zero; during the swing phase the acceleration is double integrated. Our algorithm detects the stance phase of each step by applying a threshold to the product of the norm of the acceleration by the norm of the rate of turn as previously suggested [34]. If this product is below an empirically determined threshold for more than 0.2 seconds then a stance phase is detected. When the product rises above the threshold, again a swing phase is detected. This is illustrated in Figure 6.6. If some steps are taken at a faster pace then the stance phase may not always be detected and some opportunities for ZUPTs will be missed.

In all of our experiments we sampled orientation and inertial data at 100 Hz, which is the maximum speed at which the onboard processor can compute orientation. Our algorithm also performs with similar results at 50 Hz.

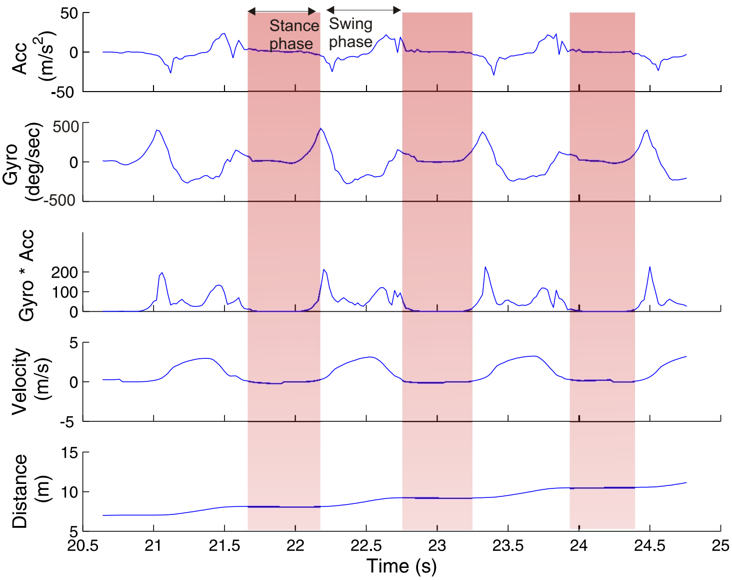


Figure 6.6: *PDR algorithm: each step has a stance phase and a swing phase. Velocity is reset to zero during the stance phase, and acceleration is double integrated during the swing phase.*

6.5.2 PDR Evaluation

In this subsection we report the performance of our PDR algorithm with experimental data gathered from different environments and for various trail topologies.

Experimental setup

In all the experiments the IMU was firmly attached under the laces of the user's shoe. We had an on-line implementation of the algorithm described in Section 6.5.1, recording the user's trajectory on a hand-held computer connected to the IMU. One set of experiments was run in a university building (Lancaster University's, Infolab). We considered different trail topologies: straight line (88 m in total, Figure 6.7a), L-shaped (54 m in total, Figure 6.7b) and rectangular (11.5 m in total, Figure 6.7c). These paths were perambulated by two users. Another set was run in a similar building in another institute (TZI, Bremen). We walked along a long corridor, entering several offices along the way (140 m in total, Figure 6.7d). A third set was run in a large industrial workshop. A single user walked a complex path of over 200 m around heavy machinery (Workshop Bremen, BIBA1, Figure 6.8). A final set was run in the office corridors around the workshop (Workshop Bremen, BIBA2, 220 m in total, Figure 6.7e. These paths were perambulated by six different users, three times each. In all the experiments the user returns to the starting point. The recorded paths shown here are each a typical example from a particular set of experiments.

Error analysis

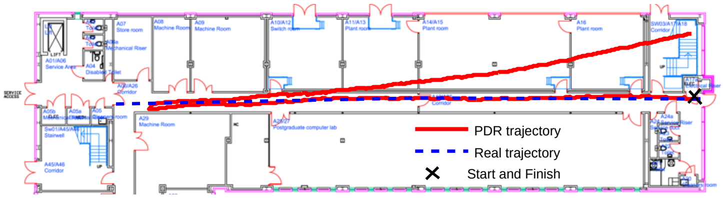
We observe two major sources of errors in the PDR approach — error in distance and error in heading. The error in distance and heading together will lead to a large error in the position (as can be seen in Figure 6.7). For the straight line in Figure 6.7a, the estimated distance drift is +2 percent of the total travelled distance. For the L-shaped path in Figure 6.7b we get an error of -8 percent of the total travelled distance. For the rectangular path in Figure 6.7c, we get a closed loop where the starting and ending points are the same, but the error is -7 percent of the total travelled distance. We notice that heading errors tend to occur when the user does a 180 degree turn.

Figures 6.7d and 6.7e show that the performance of PDR can be impacted significantly by heading errors. We tested it for cases where the user walks along a corridor and enters several rooms along the way; the path shown in Figure 6.7d starts well but severely drifts off after 40 metres. The drift happens in one particular place and then again just after the 180 degree turn. For the U-shaped path in Figure 6.7e, the error in heading is extreme, due to interference from machinery in the nearby workshop. All experiments in the same corridor at BIBA2 show an almost identical error pattern, suggesting that there is significant magnetic interference in certain locations.

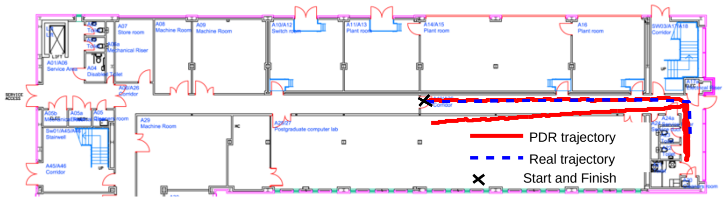
Although some distance drift is inevitable due to the integration of noise and offsets in the raw sensor data, we also believe that most of the distance error is due to the MTx incorrectly estimating its orientation as explained by Foxlin [66]. Thus we might interpret some of the forward motion as vertical motion, or vice-versa. Since MTx is a commercial product we have very little information about how the different sensors are used in computing the orientation, and almost no control over any of the internal parameters. Based on our experiments in different environments, we assume that most of the heading errors are due to metallic objects or magnetic fields interfering with the MTx magnetometers since these extreme heading errors occur systematically, in the same locations. We also note that when using the system outdoors in an open space the results are much better and the orientation drift is negligible. So the magnetometers help in outdoor situations where they accurately determine magnetic north but indoors they can cause heading errors.

6.5.3 Impact on guidance

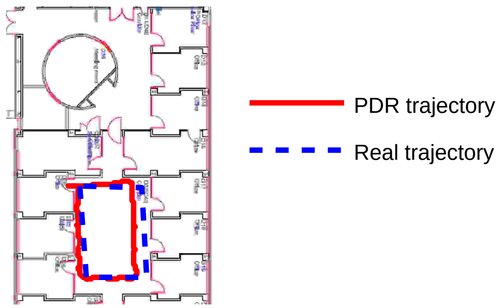
We consider the consequences of these observations on guiding a user along a path. The biggest problem we observe in the PDR approach is the drift in orientation. Even if the position is corrected by some other sensor modality, an error in the heading implies that we cannot guide the user because we do not know which direction they are facing. Drift in the distance estimates are unavoidable but they remain small and consequences for guidance are less important. The type of errors we have observed make it difficult to quantitatively evaluate performance which can vary from "almost perfect" to "unusable" depending on the level of magnetic interference. In all the cases, we observed that most individual segments of the recorded paths are very accurate — even a spiral staircase (Figure 6.8) — but that a strong heading error occurs at particular locations. Manual correction of the position and



(a) PDR straight line path, Lancaster University's Infolab

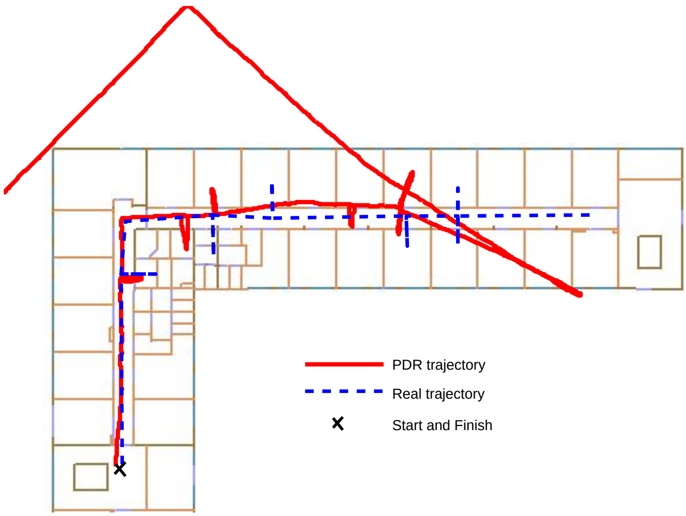


(b) PDR L-shaped path, Lancaster University's Infolab

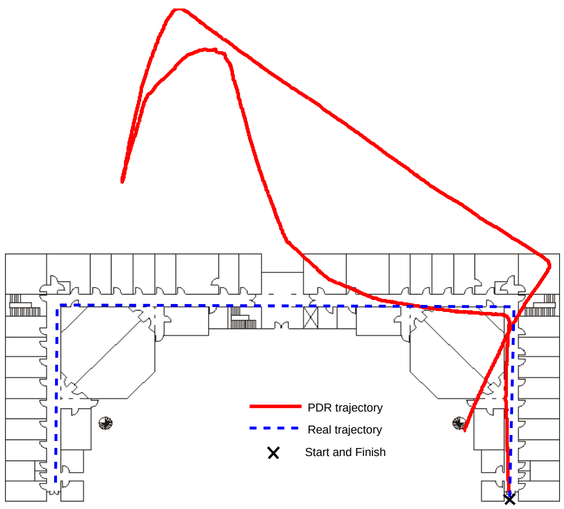


(c) PDR rectangular-shaped path, Lancaster University's Infolab

Figure 6.7: *Results of PDR for different trail topologies (continued on next page).*



(d) PDR path with user entering offices along the corridor, Bremen Insititute, TZI



(e) PDR path “worst case scenario” with strong magnetic interference due to nearby machinery, Workshop Bremen, BIBA2.

Figure 6.7: *Results of PDR for different trail topologies (continued from previous page).*

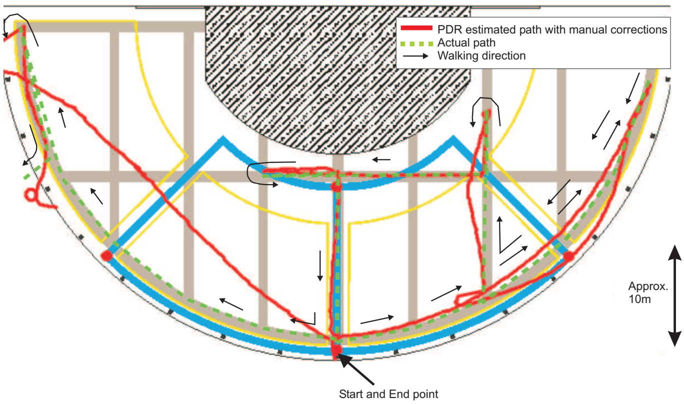


Figure 6.8: *Workshop Bremen, BIBA1, with manual correction applied to position and heading at each turn.*

heading at each turn can give good results as shown in Figure 6.8. The challenge is to make these corrections automatically.

6.6 Simulation of a guidance system

We plan for search and rescue teams to deploy small ultrasound beacons as ad hoc landmarks along their path. These beacons can then be used by other teams or by the same team to assist them on their way back. The team members wear boots equipped with ultrasound transmitters (as shown in Figure 6.2b) that can be located by the beacons, and inertial sensors. In this section, we investigate how such a system might perform through simulations.

6.6.1 Measurement model

We model the ultrasound and inertial measurements based on our observations of data from deployments in realistic environments.

Ultrasound measurement model

The ultrasound location estimates are very noisy. We model the range and bearing measurements as Gaussian with standard deviations of 5 cm and 30 ° respectively [82]. A fraction of the range measurements are large outliers. Because the ultrasound location estimates are so noisy we only use them to correct the PDR estimates if the discrepancy between the ultrasound and PDR estimates is greater than a threshold (on the scale of a metre or more). If the PDR location estimate and the ultrasound location estimate are consistent then we continue to rely on the PDR since this will give smoother results. If the estimates are not consistent then we trust the ultrasound location estimate. The ultrasound location estimate is used as the new location and the heading of the PDR is adjusted using a simple trigonometric formula which returns the angle between the current (wrong) location estimate, the last reliable location estimate, and the new (presumably more accurate) ultrasound location estimate. This relation gives good results in practice, but only if the ultrasound measurements are frequent enough. For instance, when we placed ultrasound beacons only at the corners in a path, the users are out of ultrasonic measurement range for extended periods of time.

Pedestrian dead reckoning model

The successive positions of the user are not known in advance and the error in heading is dependent on position, so it must be calculated dynamically. Based on our experimentation detailed in Section 6.5.2, we conclude that the error in heading is mostly due to magnetic interference. However, the internal Kalman filter of the inertial measurement unit essentially implies that the heading is not only affected by the local magnetic field but also by the magnetic field at previous locations. In our model we define sources of magnetic interference and for each source, a radius and an amplitude. When the user moves closer to the source than the given radius then the heading is modified by the given amplitude. The sign of the modification in heading depends on the direction that the person approaches the source. This empirical model illustrated by Figure 6.9 replicates the effects that we have observed during our experiments (as in Figure 6.7e).

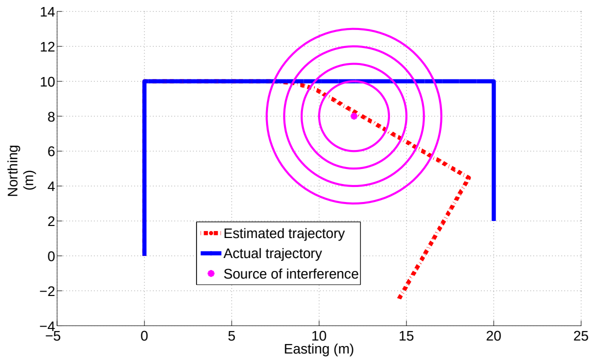


Figure 6.9: *Magnetic interference model affecting PDR estimates.*

Guidance algorithm

One important goal of our work is to guide search and rescue personnel to follow a predefined path. In practice it could be a path that was walked previously. The initial scenario we envisage is a wide open area such as a dark underground parking lot or an empty smoke-filled warehouse where a path has already been defined as the team went in and deployed ultrasound nodes along the path. As the team attempts to return to the exit back along the path they are guided by an arrow on a HMD, showing them which way to walk. The path to follow is defined as a series of segments. Given the estimated position of the user we find the point on the path that is closest to their estimated position by projecting the estimated position onto the successive segments of the path. Then we direct the user to a point that is a few metres ahead along the path as shown in Figure 6.10.

In order to check the feasibility of this system we assume that the user always follows

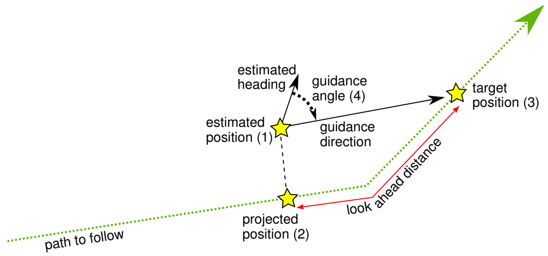


Figure 6.10: *Guidance algorithm: (1) estimate the person's position, (2) project onto path, (3) find target position, (4) compute guidance angle.*

the direction provided. This shows us how often they reach their destination and how often they stray too far from the path and get lost. The simulator is event-based: ultrasound and inertial measurements are generated periodically (every 200 ms and 10 ms, respectively) and processed by the fusion algorithm to estimate the user's position. Periodically (every 2 s) the guidance system computes which direction the user should travel and the user takes a step in that direction, effectively creating a feedback loop. A simulation run is considered successful if the person gets within a short distance of the end of the path.

6.6.2 Results and Evaluation

If we run a simulation with PDR alone, that is without using the ultrasound measurements to correct position and heading, the user will be guided to the wrong location. In the sample simulation shown in Figure 6.11 the PDR wrongly estimates the person to be too far South and so they are guided towards North. The simulation ends without the person reaching the destination because the system wrongly believes that they are already there. However, if ultrasound measurements are used to correct the position estimates, the person is successfully guided to the end of the path as shown in Figure 6.12.

Initial results show that if we do not use ultrasound measurements enough, the user will be guided away from the path and out of range of the beacons due to incorrect position and heading estimates. If that happens in the simulation, the user is lost unless by chance they stray back into range of the beacons. In reality, new nodes could be automatically deployed to create a new branch in the path or some special action could be taken if this occurs, at the very least by warning the user. If we use ultrasound measurements frequently then the user is likely to reach the end of the path safely.

In another batch of simulations, we introduce different levels of uncertainty to the ultrasound beacon positions and orientations and see how this affects the success of the guidance system. We discover that even for large errors in the estimated beacon positions the user can still reach their destination. Errors in the estimated beacon orientation are even less important as long as the user's estimated position does not drift too far from their real position. This is good news as it means that the requirements for locating the beacons should be achievable. As expected, improving the accuracy of beacon locations and orientations does improve the success rate, and thus this is one way of achieving a more reliable system. Simulations also confirm that increasing beacon range and beacon density improve success rates.

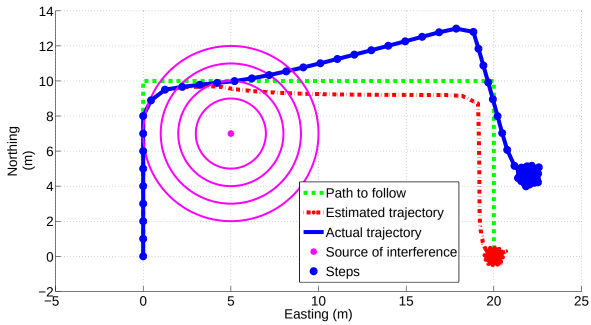


Figure 6.11: *A simulation run showing the user following guidance along a trail but failing to reach the end because of drift in the PDR estimates.*

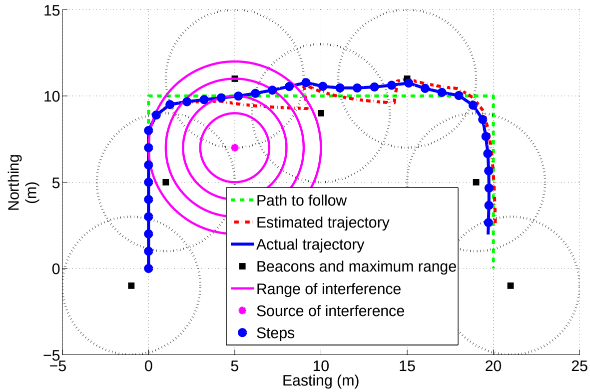


Figure 6.12: *A simulation run showing the user successfully following guidance along a trail, their position is estimated with PDR corrected by ultrasound beacons.*

The simulation results are promising and show that by using both PDR and deployed ultrasound beacons, we would be able to provide sufficient information to guide the users along a predefined path in an open space. We also showed through simulations that such a combination can even work in the presence of magnetic interference and noisy ultrasound measurements.

It is important to realise that given the life-critical field of application, such a system needs to be extremely reliable. Success rates of ninety percent are not enough or the system will not be trusted and the users will continue to rely on other navigation methods. If perfectly successful guidance proves unrealistic, then it will be necessary to investigate other ways of informing the user about the current situation, about what has gone wrong, or providing a reliable way to retreat back to a previously known position. Physically deployed beacons have the advantage of being visible, especially if the casing is carefully designed and incorporates lights or sirens, and thus can provide a very robust fallback navigation method. We have yet to study how well a real person is able to follow such guidance as provided by our system with an online implementation. We may require smoother guidance data for real users. This could be provided by more sophisticated fusion algorithms based on Kalman or particle filters.

6.7 Tracking algorithms

In this section we formulate tracking algorithms that are based on Kalman filtering. We investigate two different cases: (i) Kalman filtering of ultrasound (range and bearing) only and (ii) Kalman filtering of ultrasound (range and bearing) and inertial sensing data. In both cases we assume that the deployed nodes locations are available. The first algorithm we describe uses a Kalman filter to filter out the noise in the raw ultrasound data and by itself facilitates tracking of the mobile user following the trail of the deployed ultrasound beacons. The second tracking algorithm highlights how measurements from two different sensing media (ultrasound and inertial measurements) can be fused by a Kalman filter. The algorithms presented are inspired by single-constraint-at-a-time (SCAAT) tracking as proposed by Welch [192] where incomplete data can be used for location, as opposed to regression, or triangulation and trilateration methods where measurements are processed in batches and must contain enough information to uniquely specify a solution. The SCAAT method blends individual measurements that each provide incomplete constraints into a complete state estimate. This approach would be better-suited for sensors in a trail topology, as there are not enough constraints to solve for position when the deployed nodes are not well-connected due to the scarce connectivity along linear trails.

6.7.1 Kalman filtering of ultrasound range and bearing data

We present our use of a Kalman filter to track a mobile user along a trail of ultrasound nodes predeployed at "known" locations. The user wears an identical node on the toe of their shoe. This node emits pulses approximately five times per second. The deployed nodes hear the pulses and determine the range and bearing to the mobile node as described in Section 6.3. As reported by Hazas et al. [82] the bearing estimates are very noisy, thus using the raw measurements directly to estimate the position of the mobile node will give a rather rough trail. Smoothing the trail should have the combined benefit of making the data more pleasant to visualise on-screen (e.g. overlaid on a floor plan), making the position estimates more accurate, and enabling us at a later stage to estimate the heading of the user.

Overview of Algorithm

We formulate an Extended Kalman Filter (EKF) using a state vector $\hat{x}_k$ with four variables, two position variables u, v and two velocity variables v_u, v_v. We transform the reported range/angle measurements to Cartesian coordinates. After any discrete time step, the filter has an idea of its state and how confident it is in that state. The filter then corrects the predicted state based on the most recent measurements (range/angles converted to position) and its internal state. Figure 6.13 depicts the schematic representation of the algorithm.

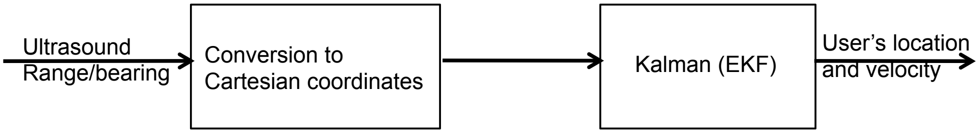


Figure 6.13: *Schematic representation of Kalman filtering of ultrasound range and bearing measurements.*

Similar to the algorithm presented in Chapter 5 we use an Iterative Extended Kalman Filter (IEKF) [144], with the difference that only one measurement per time-step is used in estimating the state vector (position and velocity of the user), as opposed to multiple measurements.

The filter is initialised with a posterior state estimate $\hat{x}_k^-$ and uncertainty P_k^-. We set the initial state estimates based on the first measurement. Each measurement consists of a timestamp, the position of the deployed node that took the measurement, and the range and bearing to the mobile node. We chose to use a small but non-zero value for P_k^-, meaning that there is a little uncertainty in the defined initial state.

We use a constant-velocity model, i.e. it is assumed the mobile node moves at constant speed between time steps (one measurement to another) but in between two measurements it is subject to additive Gaussian velocity centered on zero and of standard deviation 3 mm/s (empirically chosen). Thus, the new state estimate $\hat{x}_k$ will depend on the previous state estimate $\hat{x}_{k-1}$, constant velocity v_{xk} and a noise term w_k (as in, $\hat{x}_k = \hat{x}_{k-1} + v_{x_k} dt + w_k$). In order to predict the state using the measurements, we will have to describe how the measurements are related to the state. The *measurement model* $\hat{z}_k (= H\hat{x}_k + v_k)$ describes how measurements depend on the state estimates $\hat{x}_k$. H is the Jacobian matrix with partial derivatives of the measurement function with respect to the state $\hat{x}_k$. The measurement function here represents the range and angles converted to position estimates:

$$\begin{aligned} \hat{z}_k = [&x_b + range * cos(angle - o_b) \\ &y_b + range * sin(angle - o_b)] \end{aligned} \tag{6.1}$$

where x_b, y_b, o_b are the deployed beacons x, y coordinates and orientation and *range* and *angle* are the measured range and angles between the deployed node and the mobile node.

The predicted error covariance (P_k^-) and the state estimate ($\hat{x}_k^-$) for a time-step is given by:

$$\begin{aligned} \hat{x}_k^- &= A\hat{x}_{k-1} + Bu_k \\ P_k^- &= AP_{k-1}A^T + Q \end{aligned} \tag{6.2}$$

Here, A is the Jacobian matrix and Q is the process noise covariance.

The filter computes the posterior state estimate by taking the prior state estimate and combining with the Kalman gain K_k times the difference between the actual measurement (i.e. position) and a measurement prediction ($\hat{z}_k = H\hat{x}_k + v_k$). This is called the *innovation* or residual r. If the innovation is zero, then the predicted state estimate exactly reflects the real measurement. But if there is a difference between the predicted and the observed measurement, then the prior state estimate needs to be updated. In Eq. 6.3, K_k determines to what extent the innovation should be used in the posterior state estimation. Based on the measurement noise R and the prior error covariance P_k^-, the gain can favour the innovations or the measurements more. The measurement noise R for the position converted from range/angles is set to be equal to the measured *range* of the node.

$$\begin{aligned} K_k &= P_k^- H^T (H P_k^- H^T + R)^{-1} \\ \hat{x}_k^{i+1} &= \hat{x}_k^- + K_k(z_k - H(\hat{x}_k^- - \hat{x}_k^i)) \\ P_k &= (I - K_k H) P_{k-1}^- \end{aligned} \tag{6.3}$$

In Eq. 6.3 the Jacobian matrix H is evaluated at the most recent intermediate state estimate $\hat{x}_k^i$ (difference between the iterative extended Kalman filter (IEKF) and Extended Kalman filter (EKF)). After a number of iterations, or when the intermediate state estimate does not differ with more than a certain threshold from $\hat{x}_k^{i-1}$, the filter sets the posterior state estimate and estimates its posterior uncertainty. It is important to note that the IEKF computes the uncertainty in the state only after it finds the most accurate intermediate state estimate.

6.7.2 Kalman filtering of ultrasound and inertial data

We present our use of a Kalman filter to track a mobile user along a trail of ultrasound nodes predeployed at "known" locations using both inertial and ultrasound measurements. The user wears an ultrasound node on the toe of their shoe (as before) and an inertial sensor is attached to the foot (as shown in Figure 6.15(b)). The ultrasound node emits pulses approximately five times per second and inertial measurements are sampled at 100 Hz. The inertial measurements are recorded in "step length" i.e. the distance moved per time step and "step heading" i.e. the difference in heading between two time steps. Based on our observations reported in Section 6.5, where individual steps are not subject to huge drifts and the drift in inertial estimates occurs incrementally, we use the Kalman filtered ultrasound measurements to correct for the drift in inertial sensors.

We formulate an EKF using a state vector $\hat{x}_k$ with four variables, two position variables u, v, and two correction variables ψ and *scale*, where ψ refers to the correction factor to be applied to heading estimate and *"scale"* refers to the correction factor to be applied to distance estimates of the inertial sensors. After any discrete time step, the filter has an idea of its state and how confident it is in that state. The filter then corrects the predicted state based on the most recent measurements (range/angles converted to position as explained in Section 6.7.1) and its internal state. Figure 6.14 depicts the schematic representation of the algorithm.

The filter is initialised with a posterior state estimate $\hat{x}_k^-$ and uncertainty P_k^-. We set the initial state estimates based on the real position measurements reported by the Ubisense system deployed in the same test area. Since we use a SCAAT implementation, we order the measurements based on the recorded time-stamps as to whether the current measurement is

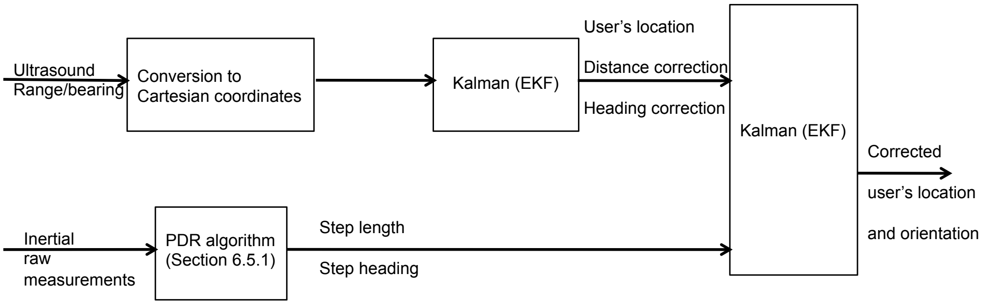


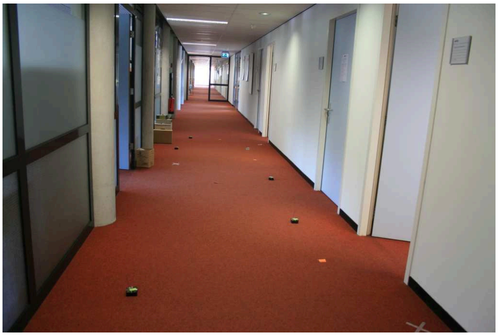
Figure 6.14: *Schematic representation of Kalman filtering of ultrasound range and bearing fused with inertial measurements.*

an ultrasound measurement or inertial measurement. Each ultrasound measurement consists of a timestamp, the position of the deployed node that took the measurement, and the relative range and bearing to the mobile node. Each inertial measurement consists of a timestamp, the step distance (difference between the current and previous position) and step heading (difference between the current and the previous heading). We chose to use a small but non-zero value for P_k^-, meaning that there is a little uncertainty in the defined initial state.

The filter operates in the following fashion. If the current measurement is an ultrasound measurement, the position and correction to be applied to the heading i.e. ψ and the error in distance is estimated i.e. *"scale"* and updated as part of the filter estimates. If the subsequent measurement is an inertial measurement, it uses the ψ and *scale* estimated by the filter when the previous ultrasound measurement was received and add this correction factor to the current inertial step length and step heading. The idea here is to correct the inertial step length and heading based on the correction factor that is estimated during the previous ultrasound measurement. Basically, when the measurement is an inertial measurement, the filter does not update the state but corrects the inertial estimates. In doing so, the drift of the inertial measurements is effectively scaled based on the correction factor determined by the ultrasound measurement. Using two different types of measurements (ultrasound and inertial) with varying sampling rate (5 Hz and 100 Hz), increases the overall update rate of the system. In principle, if the ultrasound measurements are frequent enough, the error in inertial estimates will be minimised.

6.8 Performance Evaluation of Tracking algorithms

In this section, we report the performance of the tracking algorithms that were described in Section 6.7. First, we outline in Section 6.8.1, the tesbed used for collecting data (ultrasound and inertial measurements) and the groundtruth or reference system used for the evaluation purposes. We then summarise the performance of the tracking algorithms in Section 6.8.3.



(a) Deployed nodes (in black) along the corridor

(b) Ultrasound node (blue) and Xsens inertial unit (orange) firmly attached to user's foot

(c) Snapshot of data collection performed by the user within the lab. Deployed ultrasound nodes are shown as black boxes on the floor.

Figure 6.15: *Experiment and data collection.*

6.8.1 Data collection and Groundtruth system

Twenty-one ultrasound nodes were deployed covering an area of approximately 15×9 m (refer to Chapter 5, Figure 5.2). The positions of the deployed nodes were surveyed a priori using a tape measure with reference to a wall. Eleven of the nodes were deployed in a lab which was relatively empty except for some desks along the corners of the room; three were placed in rooms along the other side of the corridor representing typical office spaces; and the remaining seven were placed in the corridor connecting the rooms and the lab. In all the experiments the inertial sensor and an ultrasound node which was transmitting ultrasonic pulses were firmly attached to the user's foot as shown in Figure 6.15(b). Along the trail-deployed nodes, measurements were gathered by two users (see Figure 6.15(c)). We considered different trail topologies as shown in Figure 6.17 – some of the path is along a corridor entering several offices along the way covering most of the deployment area, and some were within the lab. The users travelled different trajectories at least five times and occasionally stopped at some predefined points (as can be seen by a cluster in Figure 6.17) so as to see the performance over a longer period of time. We report for our analysis six such trajectories – with each path having a trace approximately seven minutes in duration.

Groundtruth or reference system

In order to validate our tracking algorithms we need to compare the results to some ground truth. We use the results of the Ubisense Location Engine [182] (refer to Section 5.8, Chapter 5 reporting the Ubisense performance) to report the groundtruth measurements. In addition to the IMU and ultrasound node, the users also carried Ubisense compact tag to gather the groundtruth for validation purposes. Although the Ubisense estimates will have some error, they are close enough for the purposes of judging the validity of the tracking performance (especially to see the shape of the trail). Additionally, since we enabled extra filtering [183] as opposed to the default Ubisense location algorithm, with no filtering, the estimates we use for comparison are more likely accurate than the results reported in Chapter 5 for the Twente set up. Figure 6.16 illustrates the effect of filtering and no filtering of Ubisense estimates with reference to reactivision camera tracking system, for the same path illustrated in Chapter 5 (Fig 5.9). Filtering has the effect of producing tighter estimates which more likely correspond to the real path. Ofcourse, the best reference system would have been to use a camera tracking system as we had in Chapter 5, but due to the limited range (thus, requiring dense network of camera's), we relied on Ubisense system as it covers much larger area (spanning multiple rooms and corridors) without having to instrument many receivers (Ubisensors). Also, for this specific application it suffices to see if the user is located correctly within a room.

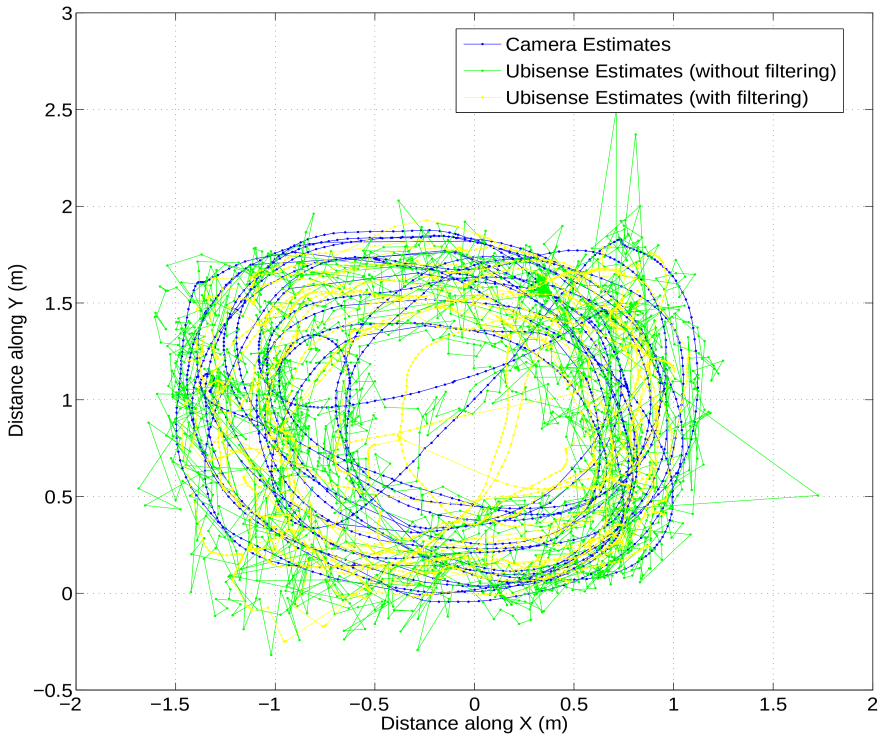


Figure 6.16: *Ubisense estimates-with and without filtering, in comparison to reactivision camera tracking system (same as Chapter 5 (Fig 5.9)).*

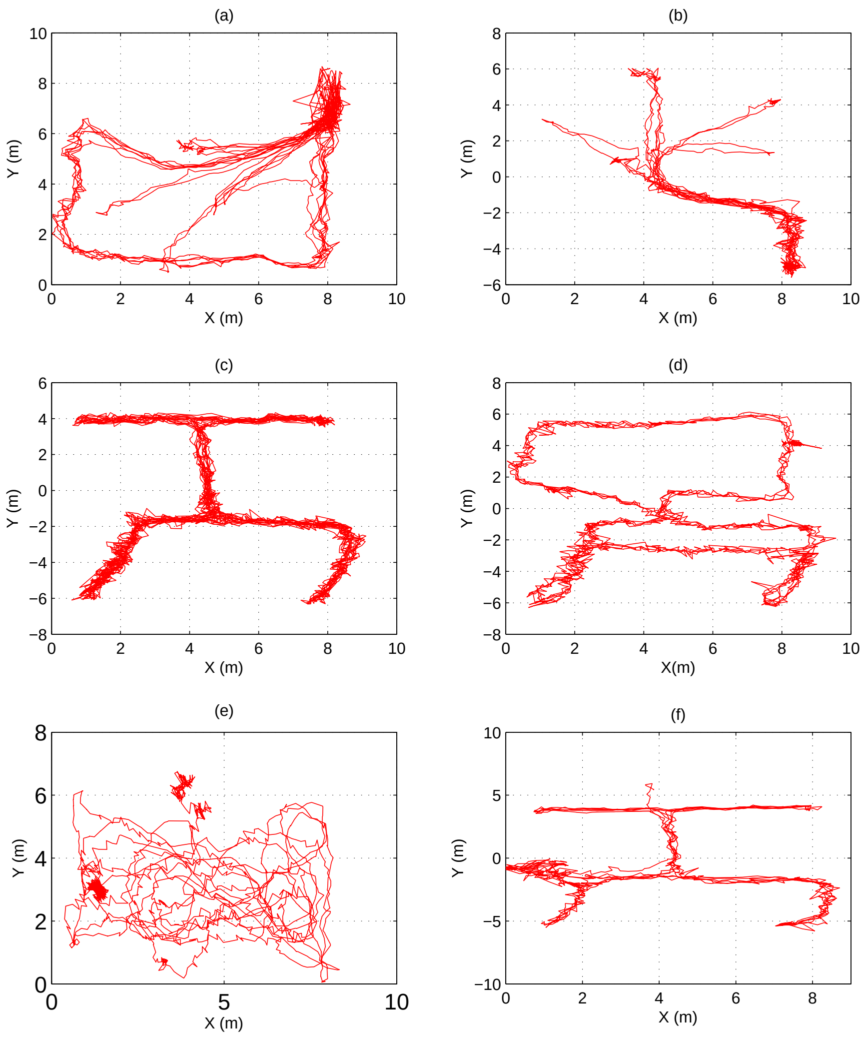
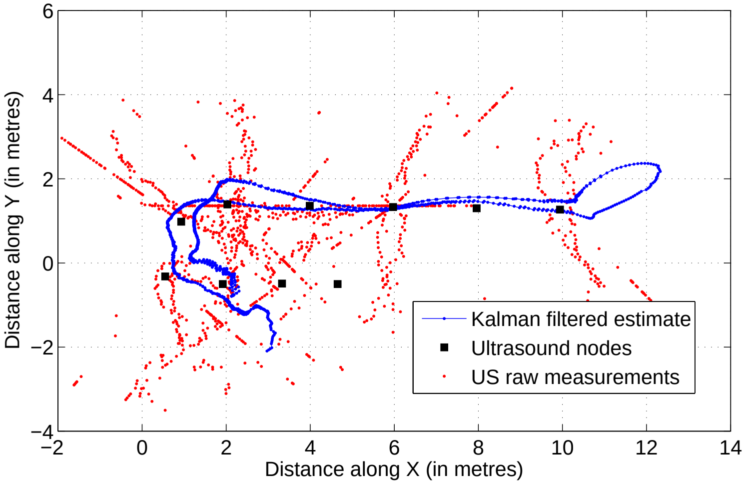


Figure 6.17: *Test path represented by Ubisense estimates (a) Square shaped path within the lab with eleven deployed ultrasound nodes, (b) L shaped path starting at the lab and walk to the room along other side of the corridor and back to the lab, with pauses at some of the points, (c) straight path walk between multiple rooms along either side of the corridor, (d) walk along the boundary of the rooms, (e) arbitrary path within the lab and (f) walk along the rooms with momentary pausing.*

6.8.2 Effect of filtering

Figure 6.18 (a) shows measurements taken with a node which was attached to the end of a stick and moved slowly along the floor of a corridor and into a lab equipped with ten deployed nodes. The "red" line indicates the raw measurements (range/angles converted to position) and plotted based on the order in which the measurements were received. The "blue" line represents the Kalman filtered trail.

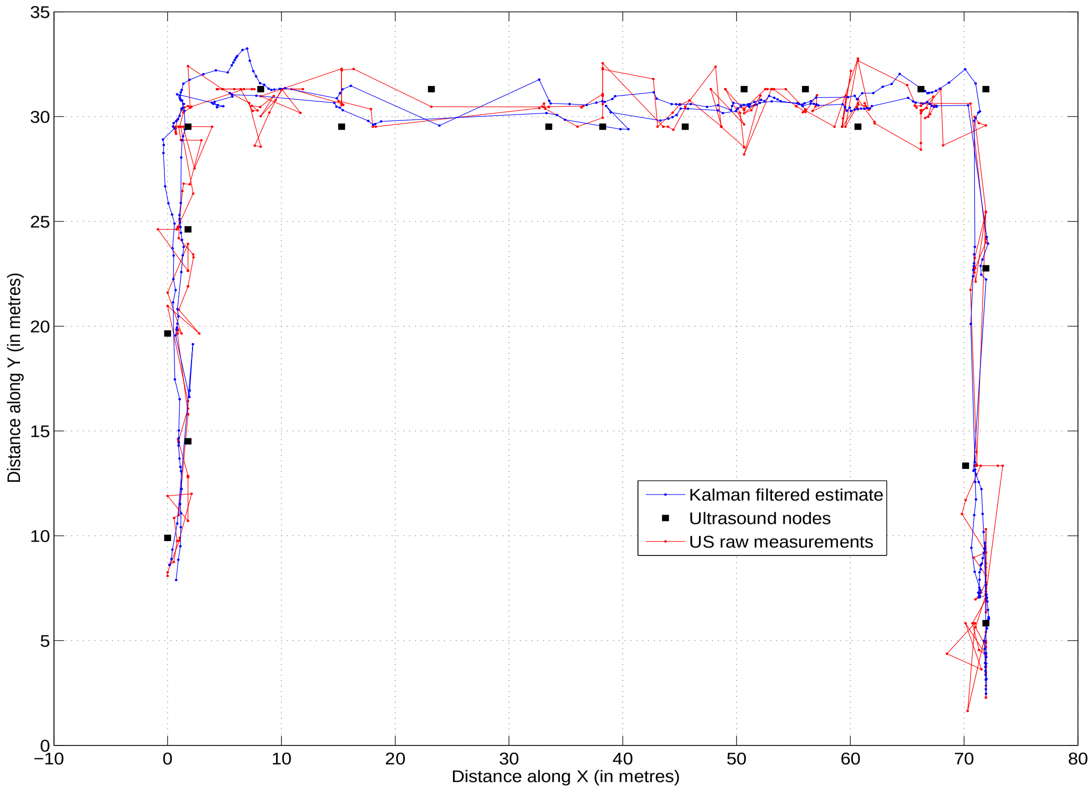


(a) Slow movement and good line of sight (experimental data). The path included walking in-between the deployed nodes.

Figure 6.18: *Unfiltered ultrasound positions and Kalman filtered trail (experimental data) demonstrating the effect of filtering on data gathered under good line-of-sight and partially blocked line-of-sight conditions (continued on next page).*

It is interesting to consider how the Kalman filtered ultrasound data look in comparison to the raw ultrasound range/angle measurements converted to position. Refering to Figure 6.18, we can observe that the filtered data is much cleaner in comparison and could almost be used alone for estimating the heading of the user. Apart from filtering the data, the Kalman filter has the added benefit of being able to generate position estimates even in the absence of measurements.

Comparing this with Figure 6.18 (b), which shows measurements taken with a node which was attached to the toe of a person's shoe, that person then walked at a normal pace along a corridor and back. Much less data was gathered during the second experiment due to the speed of the walk; this was caused by the method used for triggering the ultrasound pulses and the way that a person's leg blocks the line of sight between the mobile node and the deployed nodes. However in both cases, the filtered data is smoother than the raw data and could be used for estimating the heading of the user.



(b) Fast movement and partially blocked line of sight (experimental data), where less measurements were reported.

Figure 6.18: *Unfiltered ultrasound positions and Kalman filtered trail (experimental data) demonstrating the effect of filtering on data gathered under good line-of-sight and partially blocked line-of-sight conditions (continued from previous page).*

6.8.3 Tracking Evaluation

In this section we report the performance of both the tracking algorithms presented in Section 6.7. Table 6.1 and Figure 6.19 summarise the performance of both the algorithms by comparing fiftieth and seventy-fifth percentile errors calculated with respect to the ground truth (i.e. Ubisense estimates). Inertial combined with ultrasound performs well in all cases better than Kalman filtered ultrasound only; and accuracy is typically improved by about 1.5 m.

Figure 6.20 shows the results of tracking performance of both the algorithms (i) Kalman filtered ultrasound only and (ii) Kalman filtered ultrasound fused with inertial measurements for all six different paths that we illustrate in Figure 6.17. In general the error in both the estimates decrease periodically, confirming that the moving device was getting closer to the deployed ultrasound nodes (refer Figure 6.20). However the inertial error seems to increase gradually (Path c and Path d). This is because of the gradual drifts which we have observed in Section 6.5.

Path	**Data type**	**50% conf. level (m)**	**75% conf. level (m)**
(a)	US	2.34	4.04
	US+inertial	2.35	3.70
(b)	US	2.43	5.68
	US+inertial	2.11	4.57
(c)	US	2.78	5.23
	US+inertial	1.45	3.46
(d)	US	4.07	5.73
	US+inertial	1.06	2.38
(e)	US	2.78	4.47
	US+inertial	1.65	2.8
(f)	US	4.94	6.36
	US+inertial	3.76	4.92
Overall	US	3.25	5.24
(a)–(f)	US+inertial	1.93	3.70

Table 6.1: *Tracking performance summary. All values shown pertain to the tracking results of each of the six "walking" traces shown in Fig 6.17 (a)–(f).*

Figure 6.21 shows the estimated path of both the algorithms with Ubisense estimates plotted for the purposes of comparison. In most cases, we notice that the shape of the resultant trail matches to the Ubisense result and the PDR combined with ultrasound trail is a little smoother and tighter when compared to the ultrasound-only trail. In particular, we observe that the Ubisense results are much smoother than the presented algorithms, re-assuring that the groundtruth we have used is more likely closer to the real path. Also, from the Figure 6.21 we see that both the algorithms can perform well for the intended application of locating the user within room-level. The other benefit of fusing multiple modalities comes with regard to the update rate; since ultrasound measurements are sampled only every 5 Hz approximately while the inertial sensors are sampled at high rate (typically 100 Hz), fusing multiple modalities increases the update rate of the resultant tracking algorithm. Both algorithms' will improve if the beacon measurements are supplied more frequently.

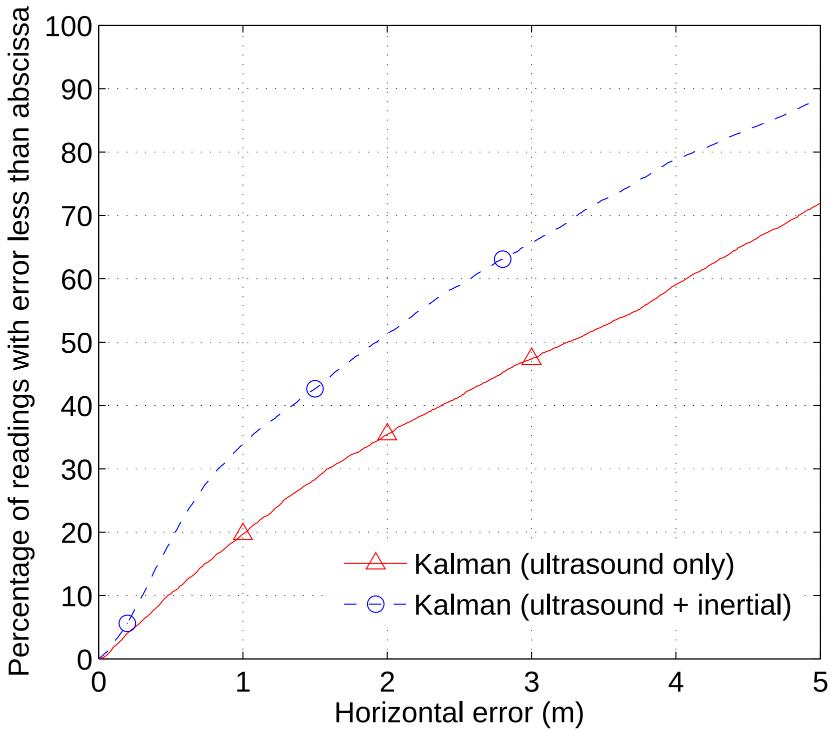
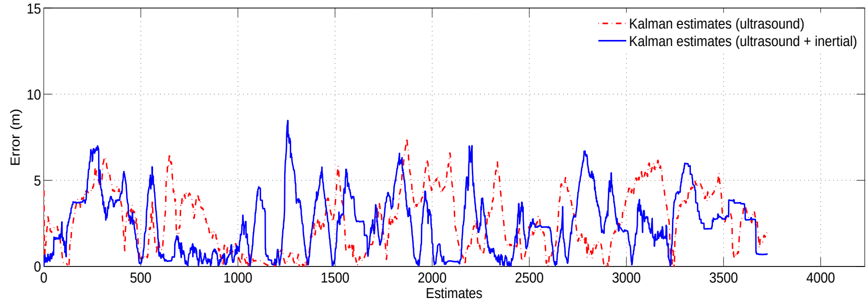
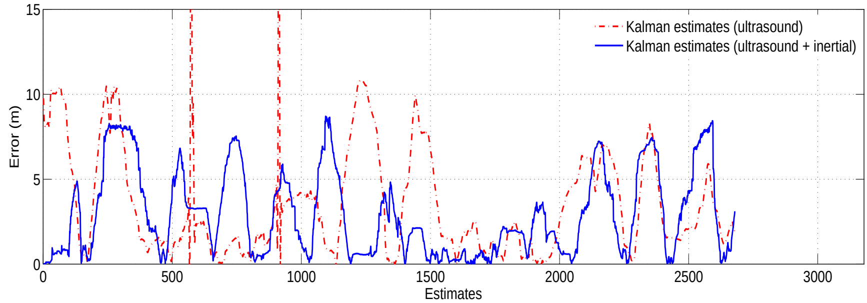


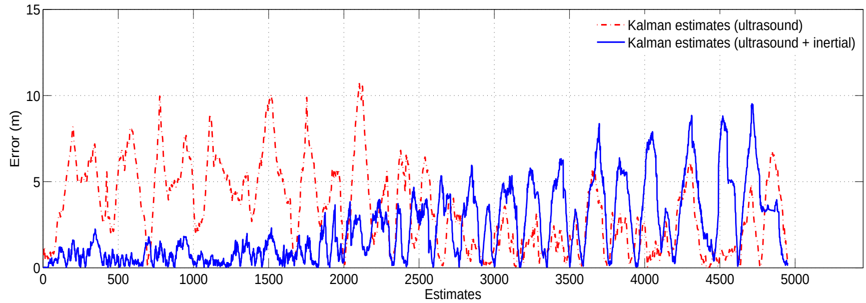
Figure 6.19: *Tracking performance of Kalman filtered ultrasound only and ultrasound fused with inertial measurements (all paths included).*



(a) Path-a error

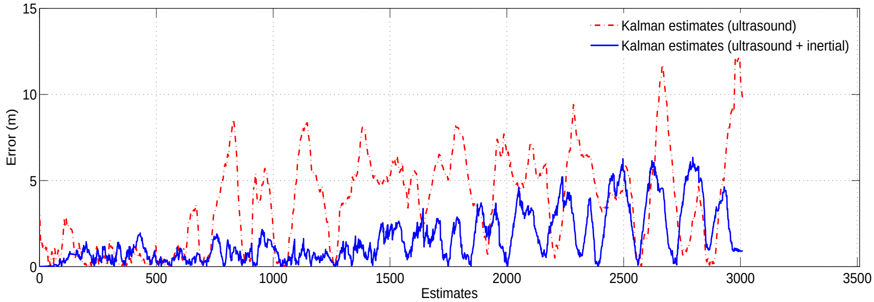


(b) Path-b error

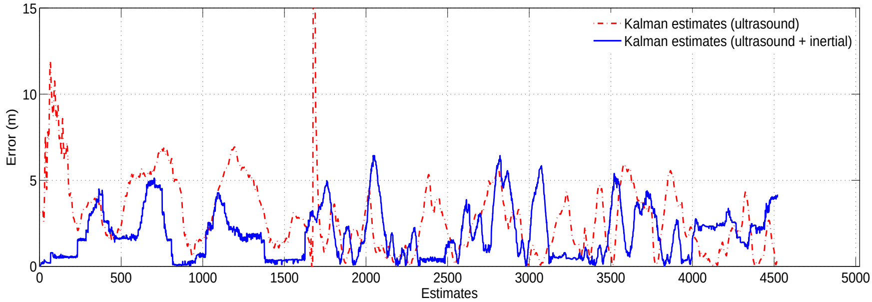


(c) Path-c error

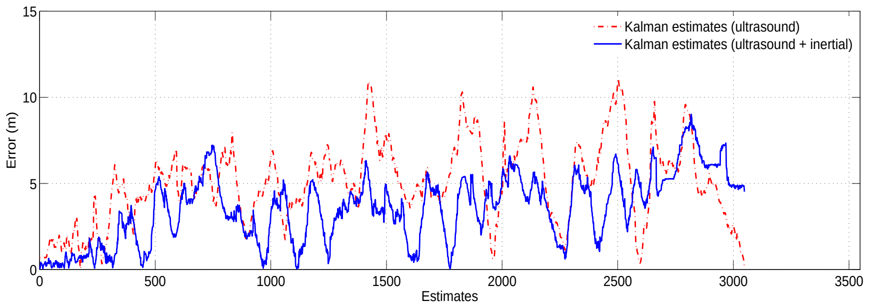
Figure 6.20: *Error with respect to Ubisense (reference) estimates of Kalman filtered ultrasound and inertial estimates and Kalman filtered ultrasound estimates (continued on next page).*



(d) Path-d error

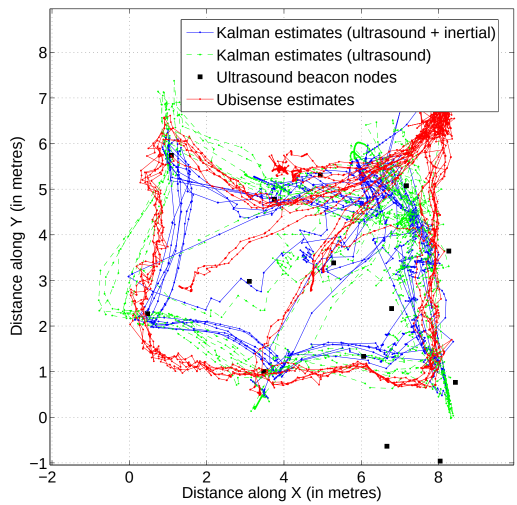


(e) Path-e error

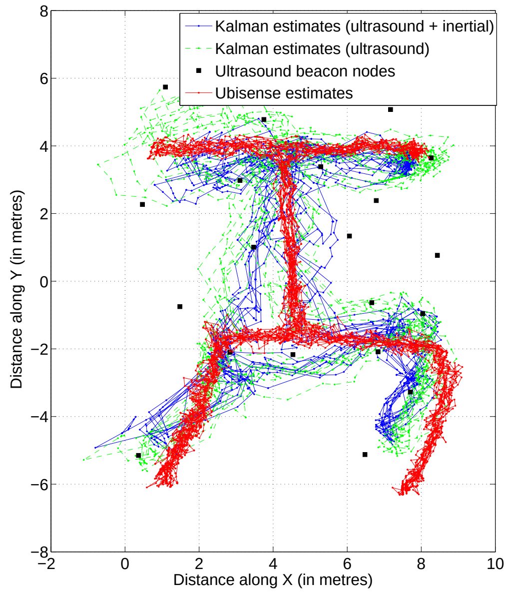


(f) Path-f error

Figure 6.20: *Error with respect to Ubisense (reference) estimates of Kalman filtered ultrasound and inertial estimates and Kalman filtered ultrasound estimates (continued from previous page).*

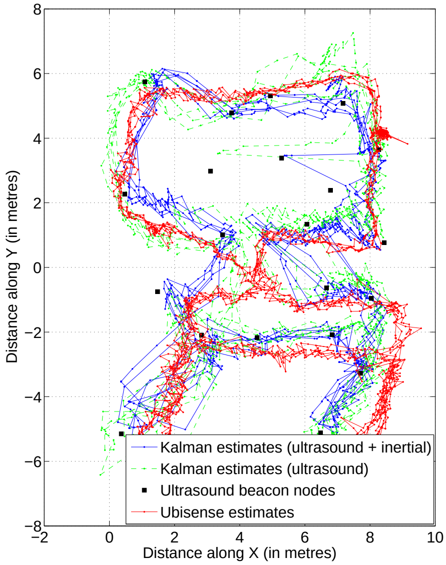


(a) Square path within a lab

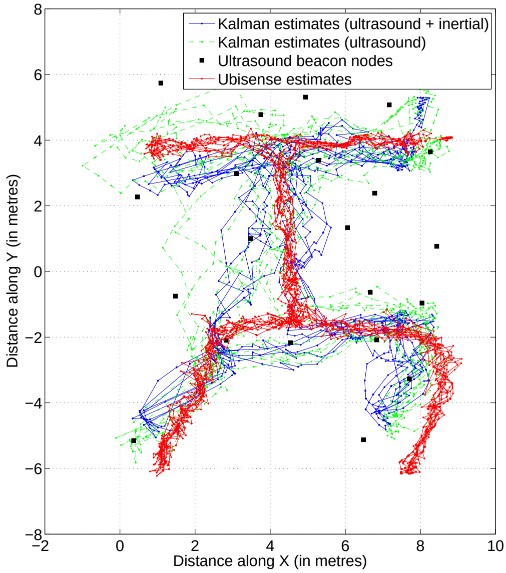


(b) Entering rooms along the corridor

Figure 6.21: *Path estimated by the (i) Kalman filtered ultrasound and inertial data combined and (ii) Kalman filtered ultrasound for different trail topologies (continued on next page).*



(c) Path with traversing along the boundaries of the room



(d) Straight Path, entering rooms along the corridor

Figure 6.21: *Path estimated by the (i) Kalman filtered ultrasound and inertial data combined and (ii) Kalman filtered ultrasound for different trail topologies (continued from previous page).*

6.9 Conclusions and Future work

In this chapter we address the use of ultrasound and inertial sensing technologies to aid firefighters by providing navigation and tracking solutions. Based on the understanding of the errors encountered in the PDR estimates, we looked into complementary technologies that can correct for the drift. Specifically, we used ultrasound sensors which have the capability to measure relative range and bearing. We showed the potential effectiveness of the combination of inertial and ultrasound measurement through a simulation of a guidance system. The simulation results of the guidance algorithm are promising and show that using both PDR and deployed ultrasound beacons to estimate a person's position we would be able to provide sufficient information to guide them along a predefined path even in presence of magnetic interference and with noisy ultrasound measurements.

We argue that smoother data is better for giving guidance in real-time to users. To assess the feasibilty of obtaining smoother data we formulated tracking algorithms based on Kalman filtering. We particularly focused on two types of tracking algorithms: (i) Kalman filtering of ultrasound range and angle measurements and (ii) Kalman filtering of inertial and ultrasound range/angle data. Both algorithms have been experimentally evaluated for different trail topologies.

Both algorithms we presented used a SCAAT-based implementation [192] of the Kalman filter, i.e. using a single measurement at a time to solve for position. The results of the ultrasound fused with inertial sensors is clearly a win over only inertial data. We also showed how ultrasound fused with inertial measurements have improved accuracy over Kalman filtered ultrasound-only measurements. This is because the error in inertial estimates starts to grow gradually, and periodic corrections from ultrasound estimates will help in minimising the drift. The fusion of ultrasound combined with inertial not only improves the accuracy, but will increase the update rate of the overall system. However, the increased processing would also consume more of the mobile device's resources. Analysing the tradeoffs is a subject of future investigation.

While we have used SCAAT-based algorithms, other types of algorithms that use a batch localisation approach which considers all the data collected by the mobile node or usage of other Bayesian methods such as particle filtering will be a subject of future investigation.

The work presented here is based on the assumption that the deployed ultrasound beacons location are known. A major challenge is calibrating the positions of the ultrasound beacons. We plan to address this as a simultaneous localisation and mapping problem (SLAM). This is a common topic in robotics but due to the nature of movement in pedestrian navigation and the trail topology of the beacons the solutions will be different.

CHAPTER VII

Conclusions

At the core of ubiquitous computing is context awareness, the concept of sensing and reacting to dynamic environments and activities. Location is a crucial ingredient of context and much research in the past decade has focused on location-sensing technologies, location-aware computing and location-based applications. This thesis focuses on formulation of *localisation algorithms* with the capability of fusing readings from multiple modalities. In Chapter 1 we stated that the main research question of this thesis was to investigate ways to bring performance improvement by incorporating multimodal data.

Although there are many performance parameters, the core assessment criterion for any location system is *accuracy*. In this thesis we have looked into a variety of ways to improve accuracy. The methods range from simple smoothing and filtering to sophisticated fusion and tracking. The simplest smoothing and averaging techniques provide modest improvements. On the other hand, Kalman filtering-based approaches offer the ability to fuse readings across different sensing technologies and incorporate motion models to improve accuracy significantly. Kalman filtering also works consistently well, amidst the absence of sensor readings, thus allowing tracking capability. Fusing multiple modalities not only improves accuracy, but also improves other desired properties like update rate, reduction in beacon density, increasing coverage, etc.

7.1 Contributions

We summarise here the main contributions and conclusions of the work presented in this thesis.

Taxonomy and survey of location systems

In Chapter 2 we have reviewed the basic principles of localisation and the classification of the state of the art based on our taxonomy. From our detailed survey, it was evident that no location system is error free and suited for all situations. For example, pure inertial sensors suffer from drift, ultrasound sensors require clear line of sight and magnetic sensors are affected by ferromagnetic and conductive materials in the environment. Thus, we rationalised *multimodal localisation* as one of the promising ways to improve location accuracy. Apart from improving performance of the location system in limited measurement volumes, fusion of heterogeneous sensing systems will ultimately allow people to move from place to place without loss of location knowledge, thus minimising the accuracy vs. coverage tradeoff. We have applied our taxonomy in Chapter 3 to choose an appropriate location sensing technology that meets the needs of a specific application.

Characterisation of raw sensor data

We have performed a detailed characterisation of a wide variety of measurements that are typically used for localisation. Thorough understanding of the behaviour of signal strength captured from WLAN enabled us to develop several motion inferencing algorithms. The raw error characterisation of individual sensing modalities (pseudoranges and angles) helped us to comprehend the actual benefits of fusion and to design "controlled" algorithms that work with both perfect and less than perfect data. Additionally, knowing the individual data characteristics together with their strength and weakness, enabled us to make appropriate choices in selecting different modalities (drift in inertial sensors corrected by ultrasound range measurements) for improving the accuracy. The other benefit of characterisation is that the error distributions required by the positioning algorithms to function efficiently can be derived by having a closer look at the underlying data. All our analysis is performed on sensors and technologies (either commercially available or as a research prototype) that are being extensively used by the wider community. So at large, thorough characterisation of common measurements like what we have presented in this work can serve as a guideline in understanding the characteristics of some of the demanding location technologies that exist today.

Algorithms for inferring motion and location from WLAN RSSI

In Chapter 4, we have presented novel algorithms for inferring the movement of a device based on observing the fluctuations in the signal strength gathered from WLAN access points. Our motion detection algorithm that is based on frequency domain analysis reports a precision and recall over 90% in distinguishing "still" and "moving" states. We have demonstrated how commonly used location algorithms like Centroid and Weighted centroid could benefit from knowing the motion of the device to be located and by using the history of past location readings to improve accuracy. The solution we have provided is smoothing of location estimates based on motion derived from RSSI. We have evaluated the performance of algorithms against traces of RSSI data collected from different environments. Our results show that addition of motion derived from RSSI provides modest improvements. Incorporating map-matching methods and/or data from inertial sensors, in combination with probablistic methods like particle or Kalman filtering might be a suitable venue for future research.

Positioning algorithms using heterogeneous data

Chapter 5 and Chapter 6 have demonstrated the benefits of fusion and tracking on sophisticated data such as the TOA, AOA and TDOA measurements gathered from some of the most demanding technologies that are in use today, namely: ultrasound, inertial and ultra-wideband.

- Specifically, in Chapter 5 we have addressed the benefit of heterogeneous observations (pseudoranges and angles) gathered from a commercially available ultra-wideband system. We have presented positioning algorithms that are based on an error minimisation approach (non-linear regression) and a state-estimation approach (Kalman filtering) using heterogeneous data collected from two very different deployments – mimicking a real-world deployment vs. an ideal deployment. We demonstrated that the presented algorithms work both with perfect and imperfect data and highlighted the impact of calibration on accuracy of the location estimates. For accurate raw data, the algorithms exhibit similar performance, and we would select the Kalman filter since it provides a more consistent (if at times slightly less accurate) stream of location estimates. Supplied with reliable readings, the Kalman filter performs better than regression as deployment density decreases. For deployments with poor calibration and/or less reliable,"noisy" readings, we would select the non-linear regression algorithm for its accuracy, despite the high ratios (20–80%) of rejected readings. We have shown that the algorithms can work well on homogeneous data (pseudoranges or AOAs), despite the reduction of the information contributing to the location solution. Under certain configurations, noisy, homogeneous data can be pathological. When working on homogeneous data with reliable accuracy, the algorithms continue to produce good location estimates.

- In Chapter 6, we have addressed the use of ultrasound and inertial sensing technologies for providing navigation and tracking solutions. Based on the detailed characterisation of the errors in dead reckoning estimates we looked into complementary technologies that can correct for the drift. Specifically, we have used ultrasound sensors which have the capability of measuring ranges and angles. We have showed the effectiveness of the combination of inertial and ultrasound measurement through a simulation of a guidance system. We envisaged that smoother data is required for guidance purposes. To see the feasibilty of obtaining smoother data we formulated tracking algorithms using a SCAAT-based Kalman filtering approach. We particularly focused on two types of tracking algorithms: (i) Kalman filtering of ultrasound range and angle measurements and (ii) Kalman filtering of inertial data corrected by ultrasound range/angle data. Both the algorithms have been experimentally evaluated for different trail topologies. Looking at the results of the ultrasound-aided dead reckoning solution, it is clearly a win over using only inertial data. We also showed how ultrasound fused with inertial measurements outperforms ultrasound-only tracking. The fusion of ultrasound with inertial measurement, not only improves the accuracy, but will also increase the update rate of the overall system. However, the increased processing would also consume more of the mobile device's resources and is a subject of future investigation to analyse the tradeoff.

7.2 Concluding remarks

At the very beginning of this thesis we hypothesised that regardless of the type of measurements, fusing multimodal information would help in bringing performance benefits. We have formulated location algorithms that operate on a wide variety of measurements ranging from simple and easily available RSSI data to complex timing and angles of arrival information obtained from different types of technologies and proved that our original hypothesis was indeed valid.

The presented algorithms in general make use of knowledge of *motion* to improve their estimates. In Chapter 4 we have demonstrated the effect of smoothing by incorporating the knowledge of the device movement with a commonly used deterministic algorithm. While the presented algorithm in Chapter 4 is different than the one presented in Chapter 5 and Chapter 6, we see some commonalities between the algorithms presented in Chapter 5 and Chapter 6. Essentially we have used Kalman filtering for performing different functionalities – *fusion*, *filtering* and *tracking*. The Kalman filter assumes that the system being estimated has a measurement equation which describes how the measurements are related to the state to be estimated. If the system being estimated has multiple forms of observation, there would be multiple corresponding measurement equation, each representing a different relationship. The filter can then effectively combine or fuse information contained in the heterogeneous measurements. The benefit of fusing two heterogenous observations (pseudoranges and angles-of-arrival) from an UWB system has been detailed in Chapter 5. In Chapter 6, we have presented tracking algorithms based on Kalman filtering. The first tracking algorithm we describe uses a Kalman filter for *filtering out* the noise in the raw ultrasound data and by itself facilitates tracking of the mobile user following the trail of the deployed "known" ultrasound beacons. The second tracking algorithm highlights how ultrasound range and angle measurements can *correct* for drift in inertial estimates using a Kalman filter. Although improving accuracy was our main focus, we also listed in brief how other desired properties such as increasing update rate, coverage and reducing the infrastructure can be achieved by multimodal localisation.

Evaluating a location algorithm is rather difficult and can be considered as a chicken-and-egg problem. We need to have precise groundtruth to be able to compare the errors in the estimated position. We have tried our best that a modest evaluation can be made at each stage of our work. We employed four different techniques for evaluating different parts of the work presented in this thesis: (i) a combination of conventional tape-measures and Total Station for static performance analysis, (ii) camera-based tracking was used to validate the UWB positioning algorithms for assessing the tracking performance, (iii) the Ubisense system was used for evaluating the tracking performance of Kalman estimates of inertial and ultrasound sensors and (iv) for WLAN localisation algorithm evaluation we have used manual annotations in combination with an interpolation algorithm to generate the groundtruth of a larger area.

Extensions to the presented work

While the focus of this work was towards formulation of algorithms that are capable of fusing multimodal data and in comprehending the benefits by applying the concept of multimodal localisation on a large variety of measurement types, it would be interesting to analyse the tradeoffs in terms of accuracy vs. complexity and extending the presented algorithms to provide real-time support. This would be be an important step especially, when the presented algorithms have to be scaled down in order to cater for providing real-time support in resource-constrained mobile platforms. However, with increasing improvements in the mobile platforms, this issue might fade away in due course of time.

In general, we remain a little curious about how slight changes in the model parameters would affect the Kalman filters performance presented in Chapter 5 and Chapter 6. Although we have chosen the parameters based on our detailed analysis of understanding the error distributions of the measurements, while monitoring the progress of parameter search, we did notice that large improvements i.e. error reductions can be achieved with some parameter changes, and some parameters did not impact the results at all. It would be interesting to dig a little deeper to evaluate the sensitivity of a particular system to changes in the dynamic model parameters. In future, we plan to explore *adaptive Kalman filters* that have the capability of tuning the thresholds automatically based on the current measurement.

We have looked into the usage of some of the algorithms that have the capability to fuse data from multiple modalities. But, there exist a lot of other algorithms which can operate on multimodal data. In this respect, there is much to be learned from algorithms designed by the robotics community.

7.3 Further Research

Looking back from the days when early navigation systems were invented, decades of research have made localisation evolve to a mature stage. This does not however imply that localisation is a solved problem. The "holy-grail" system that can work at any time, anywhere, offering good accuracy, high coverage, low cost and preferably self-contained positioning capability is still not achieved yet, especially for indoor localisation. While developing a "holy-grail" in our opinion is hard, we strongly believe that the research community will still strive hard, until the thirst of achieving high accuracy-low cost-high coverage is quenched.

Looking ahead, mobile devices will continue to shrink in size and price while offering more capability and usefulness to people. This will increase the interaction that users have with location-aware devices and services in future. The increasing density of a wide variety of wireless networking and the new location fusion algorithms hold the promise of hybrid location systems, that are indeed minimising the accuracy, coverage and cost (both infrastructure and calibration) tradeoff's. Advances in location sensing technologies and factors that are promoting wide-scale coverage are making coarse-grained location information widely available. Over time, fine-grained location systems and applications will become more economically viable and easily deployable. However, a number of issues such as privacy, sensor fusion algorithms, autocalibration aspects, cheap and self-contained location system still represent fertile research areas and have to be looked into in more detail in order to realise wider public acceptance of the location systems and to minimise the accuracy, cost and calibration tradeoffs.

We have already highlighted the impact of calibration on accuracy of the estimates. In most of the location systems, knowledge of beacon position or in some cases orientation is to be known a priori for facilitating location estimation. These parameters can be mapped in offline (under controlled circumstances) and mostly once prior to use. However, in some cases where the positioning system has to be deployed ad hoc, an assumption of beacon positions knowledge may not be applicable. Instead it might be desirable to estimate some of these parameters dynamically while the system is operational. This makes the idea of *autocalibration* attractive. The algorithms presented in both Chapter 5 and 6 assumes the availability of the location and orientation of the beacons. Manual approaches of measuring the beacon position and orientation are time consuming and error prone. The most effective method of calibration would be one that is performed automatically. Algorithms presented in both Chapter 5 and Chapter 6 could be extended to address *autocalibration*. While there is work already being carried out in this topic in virtual tracking [192], robotics [144] and mobile and ubiquitous computing [55], positioning and autocalibration systems have to be combined effectively as one package for realising accurate yet easily deployable location systems.

As a final remark, the development in indoor positioning technologies, with parallel progress in reduction of the costs, will spread the indoor positioning to more and more facets of life. It is hard to predict the amount of innovation that wide-spread indoor positioning, in combination with other technologies will enable in the future. It is possible to foresee a time when everyone knows exactly where they are, wherever they are, at all times and the applications that will arise from the availability of that information will continue to surprise us!

Bibliography

[1] Active Campus. www.activecampus.ucsd.edu.

[2] Aeroscout. http://www.aeroscout.com/.

[3] Air Location II. http://www.engadget.com/2006/10/02/.

[4] Ascension Technology Corporation. http://www.ascension-tech.com/.

[5] AT&T Friend Finders. http://www.wireless.att.com/cell-phone-service/welcome/index.jsp.

[6] Bluesoft inc. www.directionsmag.com/companies/Bluesoft,_Inc./.

[7] CeBIT. http://www.cebit.de/news.

[8] Crossbow Technology, Mica 2 Mote. http://www.xbow.com/Products/productdetails.aspx?sid=174.

[9] Dodgeball.com:Social Mobile Software. www.dodgeball.com.

[10] Garmin: World leader in sat-nav. http://www.garmin.com/garmin/cms/site/uk.

[11] Idtechex. Reports. http://www.idtechex.com/research/reports/item_level_rfid_2008_2018_000202.asp.

[12] Intel Place Lab. http://www.placelab.org/.

[13] ISO/IEC 18000-7:2004. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=37978.

[14] Legoland finds lost kids with RFID tracking. http://www.networkworld.com/includes/ads-pre2.html.

[15] ntag Interactive. www.ntag.com.

[16] Polhemus Innovation in Motion. http://www.polhemus.com/.

[17] Skyhook Technologies. http://www.skyhookwireless.com/.

[18] Spot me. http://www.spotme.com/.

[19] Time Domain Corporation. http://www.timedomain.com/.

[20] Tom Tom : The smart choice in personal navigation. http://www.tomtom.com/.

[21] Relate website. http://eis.comp.lancs.ac.uk/relate, 2006. Accessed 2008.01.08.

[22] RTLS Market Overview. http://rfidwizards.com/index.php?option=com_content\&task=view\&id=378%&Itemid=344, 2006.

[23] AT&T Getting First Responders on the Same Wavelength, 2007. Accessed 2008.01.11.

[24] OpenNETCF Consulting LLC, Smart device framework. http://www.opennetcf.com/Prodcuts/SmartDeviceFramework/tabid/65/Default.aspx, 2007.

[25] Ultra-wideband platform published as ISO and IEC standard. http://www.wimedia.org/imwp/idms/popups/pop_download.asp?contentID=10832, March 2007.

[26] Mike Addlesee, Rupert Curwen, Steve Hodges, Joe Newman, Pete Steggles, Andy Ward, and Andy Hopper. Implementing a Sentient Computing System. *IEEE Computer Magazine*, 34(8):50–56, 2001.

[27] Mike Addlesee, Alan Jones, Finbarr Livesey, and Ferdinando Samaria. The ORL Active Floor. *IEEE Personal Communications*, 4(5):35–41, October 1997.

[28] I.F. Akyildiz, W.Su, Y.Sankarasubramaniam, and E.Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114, 2002.

[29] Ian Anderson and Henk Muller. Context Awareness via GSM Signal Strength Fluctuation. In the Fourth International Conference on Pervasive Computing, Late breaking results, May 2006.

[30] John R. Anderson. Abandoned Cold Storage Warehouse Multi-Firefighter Fatality Fire. Technical report, United States Fire Administration, 1999.

[31] Ronald T. Azuma. A Survey of Augmented Reality. *Presence*, 6:355–385, 1997.

[32] Paramvir Bahl and Venkata N. Padmanabhan. RADAR: An In-Building RF-based User Location and Tracking System. In

Proceedings of Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), pages 775–784, 2000.

[33] Paramvir Bahl and Venkata N. Padmanabhan. RADAR: An inbuilding RF based user location and tracking system. In *Proceedings of Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 775–784, March 2000.

[34] Stéphane Beauregard. Omnidirectional Pedestrian Navigation for First Responders. In *Fourth International Workshop on Positioning, Navigation and Communication, 2007. WPNC'07*, pages 33–36, 2007.

[35] R. Bencina and M.Kaltenbrunner. The Design and Evolution of Fiducials for the reacTIVision System. In *Proceedings of the Third International Conference on Generative Systems in the Electronic Arts (Third Iteration 2005), Melbourne (Australia)*, 2005.

[36] Urs Bischoff, Martin Strohbach, Mike Hazas, and Gerd Kortuem. Constraint-Based Distance Estimation in Ad hoc Wireless Sensor Networks. In Holger Karl Kay Rómer and Friedemann Mattern, editors, *Proceeding of EWSN 2006*, volume 3868 of *Lecture Notes in Computer Science*, pages 54–68. Springer, 2006.

[37] Philipp Bolliger, Kurt Partridge, Maurice Chu, and Marc Langheinrich. Improving Location Fingerprinting through Motion Detection and Asynchronous Interval Labeling. In Aaron Quigley and Tanzeem Choudhury, editors, *Fourth International Symposium on Location- and Context-Awareness (LoCA 2009), 7-8 May 2009, Tokyo, Japan*, LNCS, Berlin Heidelberg New York, May 2009. Springer.

[38] S. Bosch, M. Marin-Perianu, R. S. Marin-Perianu, J. Scholten, and P. J. M. Havinga. FollowMe! Mobile Team Coordination in Wireless Sensor and Actuator Networks. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications 2009, Galveston, Texas, USA*, pages 151–161, Los Alamitos, March 2009. IEEE Computer Society Press.

[39] Azzedine Boukerche, Horacio A.B.F, Oliveira, Eduardo F. Nakamura, and Antonio A.F. Loureiro. Vehicular Ad Hoc Networks: A New Challenge for Localization-Based Systems . *Computer Communications. Elsevier*, 2008.

[40] D. Brenkley, S. C. Bennett, and B. Jones. Enhancing Mine Emergency Response. In *Proceedings of Twenty-eigth International Conference on Safety in Mines Research Institutes, Sinaia, Romania*, 1999.

[41] R. G. Brown and P. Y. C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering*. Second Edition, John Wiley & Sons, Inc, 1992.

[42] B. Brumitt, B.Meyers, J.Krumm, A.Kern, and S.Shafer. EasyLiving: Technologies for intelligent environments. In *Handheld and Ubiquitous Computing, Second International Symposium*, pages 1–15, 2000.

[43] N. Bulusu, J. Heidenmann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, 7(5):28–34, October 2000.

[44] Nirupama Bulusu. *Self-Configuring Location Systems*. PhD thesis, University of California, Los Angeles, 2002.

[45] Nirupama Bulusu, Deborah Estrin, Lewis Girod, and John Heidemann. Scalable Coordination for Wireless Sensor Networks: Self-configuring Localization Systems. In *Proceedings of the Sixth International Symposium on Communication Theory and Applications (ISCTA '01)*, July 2001.

[46] P. Castro, P.Chiu, T.Kremenek, and R.Muntz. A probabilistic room location service for wireless networked environments. In *Proceedings of Ubiquitous computing*, 2001.

[47] Rick Cavallaro. The FoxTrax Hockey Puck Tracking System. *IEEE Computer Graphics and Applications*, 17:6–12, March-April 1997.

[48] A.D. Cheok, S.W.Fong, K.H.Goh, X.Yang, W.Liu, and F.Farzbiz. Human Pacman: a sensing-based mobile entertainment system with Ubiquitous Computing and tangible interaction. In *Proceedings of the Second Workshop on Network and System*, pages 106–117. ACM Press, 2003.

[49] Marc Ciurana, David López, and Francisco Barceló-Arroyo. Softoa: Software ranging for toa-based positioning of wlan terminals. In *Proceedings of the Fourth International Symposium on Location and Context Awareness (LoCA '09)*, pages 207–221, Berlin, Heidelberg, 2009. Springer-Verlag.

[50] R. Collins, A.Lipton, J.Fujiyoshi, and T.Kanade. Algorithms for cooperative multisensor surveillance. *IEEE Communication Magazine*, 89(10):1456–1477, 2001.

[51] Donna Cox, Volodymyr Kindratenko, and David Pointer. IntelliBadge. In *Proceedings of First International Workshop on Ubiquitous Systems for Supporting Social Interaction and Face-to-Face Communication in Public Spaces*, pages 41–47. ACM, 2003.

[52] Diego López de Ipĩna, Paulo R.S.Mendonca, and Andy Hopper. TRIP: A Low-Cost Vision-Based Location System for Ubiquitous Computing . *Personal and Ubiquitous Computing*, 6:206–219, 2002.

[53] Goran M. Djuknic and Robert E. Richton. Geolocation and Assisted GPS. *IEEE Computers and Graphics*, 34(2):123–125, February 2001.

[54] L. Doherty, K. Pister, and L.El Ghaoui. Convex position estimation in wireless sensor networks. In *IEEE INFOCOM*, April 2001.

[55] Paul Duff and Henk Muller. Autocalibration algorithm for ultrasonic location systems. In *Proceedings of Seventh IEEE International Symposium on Wearable Computers, White Plains, NY*, October 2003.

[56] Ekahau Positioning System. `http://www.ekahau.com`.

[57] J. Ekimoto, T.Miyaki, and T.Ishizawa. LifeTag: WiFi-based continuous location-logging for life pattern analysis. In *Proceedings of Third International Symposium on Location and Context-Awareness (LoCA), 2007*, 2007.

[58] J. Elson and D. Estrin. *Wireless Sensor Networks: A bridge to the Physical World*. Kluwer, 2004.

[59] R.F. Fahy. U.S.Fire Sevice Fatalities in Structure Fires, 1977-2000. Technical report, 2002.

[60] Carl Fischer, Kavitha Muthukrishnan, Mike Hazas, and Hans Gellersen. Ultrasound-aided pedestrian dead reckoning for indoor navigation. In *Proceedings of the First ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments (MELT 2008)*, 2008.

[61] M. Flintham, S.Benford, R.Anastasi, T.Hemmings, A.Crabtree, C.Greenhalgh, Tandavanitj, M.Adams, and J.Row-Farr. Where on-line meets on the streets: experiences with mobile mixed reality games. In *In Proceedings of the conference on Human factors in computing systems*, pages 569–576, 2003.

[62] Robert J. Fontana, , E. Richley, and J. Barney. Commercialization of an ultra wideband precision asset location system. In *Proceedings of the IEEE Conference on Ultra Wideband Systems and Technologies*, pages 369–373, Reston, Virginia, USA, November 2003.

[63] Robert J. Fontana. Experimental Results from an Ultra Wideband Precision Geolocation System. In *Ultra-Wideband, Short-Pulse Electromagnetics*, pages 215–224, May 2000.

[64] Robert J. Fontana and Steven J. Gunderson. Ultra-wideband Precision Asset Location System. In *Proceedings of the IEEE Conference on Ultra Wideband Systems and Technologies*, pages 147–150, Baltimore, USA, May 2002.

[65] D. Fox, J.Hightower, L.Liao, D.Schulz, and G.Borriello. Bayesian Filtering for Location Estimation. *IEEE Pervasive Computing*, 2003.

[66] Eric Foxlin. *Motion Tracking Requirements and Technologies*, chapter 8. Lawrence Erlbaum Associates, 2002.

[67] Eric Foxlin. Pedestrian Tracking with Shoe-Mounted Inertial Sensors. *IEEE Computer Graphics and Applications*, 25(6):38–46, 2005.

[68] Eric Foxlin, Michael Harrington, and George Pfeifer. Constellation: A wide-range wireless motion-tracking system for augmented reality and virtual set application. In Michael Cohen, editor, *Proceedings of SIGGRAPH 98*, pages 371–378. Addison Wesley, 1998.

[69] Garmin. Garmin's devices to get uLocate's friend finder Buddy Beacon. `http://www.gpsbusinessnews.com/index.php?action=article\&numero=990`, August 2008.

[70] Lewis Girod, Martin Lukac, Vlad Trifa, and Deborah Estrin. The design and implementation of a self-calibrating distributed acoustic sensing platform. In *Proceedings of the Fourth International conference on Embedded networked sensor systems(SenSys '06)*, pages 71–84. ACM, 2006.

[71] Stanton A. Glantz and Bryan K. Slinker. *Primer of Applied Regression and Analysis of Variance*. McGraw-Hill, New York, second edition, 2000.

[72] Google. Google's Latitude. `http://www.pcworld.com/article/158909/google_latitude_service_lets_you_track_your_friends_how_it_works.html`.

[73] Duncan Graham-Rowe. Indoor 'sat-nav' could save firefighters. *New Scientist*, 196(2634):24, December 2007.

[74] David Graumann, Jeffrey Hightower, Walter Lara, and Gaetano Borriello. Real-world Implementation of the Location Stack: The Universal Location Framework. *IEEE Workshop on Mobile Computing Systems and Applications*, 2003.

[75] William G. Griswold, Robert Boyer, Steven W.Brown, Tan Minh Truong, Ezekiel Bhasker, Gregory R.Jay, and R.Benjamin Shapiro. ActiveCampus Sustaining Educational Communities through Mobile Technology. Technical report, University of California San diego, 2002.

[76] William G. Griswold, Patricia Shanahan, Steven W.Brown, Robert Boyer, Matt Ratto, R.Benjamin Shapiro, and Tan Minh Truong. ActiveCampus: Experiments in Community-Oriented Ubiquitous Computing. *IEE Computer Magazine*, 37(10):73–81, October 2004.

[77] A. Haeberlen, E. Flannery, A.M.Ladd, A.Rudys, Dan A.Wallach, and Lydia E.Kavraki. Practical robust localization over large-scale 802.11 wireless networks. In *Proceedings of the Tenth annual International conference on Mobile Computing and networking (MobiCom 2004)*, pages 70–84. ACM, New York, 2004.

[78] David L. Hall and James Llinas. *Handbook of Multisensor Data Fusion*. CRC Press, 2001.

[79] Peter Harrop, Glyn Holland, and Raghu Das. Real Time Locating Systems 2008-2018 - Assess all the technologies and opportunities. http://media2.idtechex.com/.

[80] Andy Harter and Andy Hopper. A distributed location system for the active office. *IEEE Network*, 8(1):62–70, February 1994.

[81] Homayoun Hashemi. The Indoor Radio Propagation Channel. In *Proceedings of IEEE*, volume 81, 1993.

[82] Mike Hazas, Christian Kray, Hans Gellersen, Henoc Agbota, Gerd Kortuem, and Albert Krohn. A relative positioning system for co-located mobile devices. In *Proceedings of the Third international conference on Mobile systems, applications, and services (MobiSys '05)*, pages 177–190, New York, NY, USA, 2005. ACM Press.

[83] Mike Hazas, James Scott, and John Krumm. Location-Aware Computing Comes of Age. *IEEE Computer*, 37(2):95–97, February 2004.

[84] Mike Hazas and Andy Ward. A high performance privacy-oriented location system. In *Proceedings of the IEEE International Conference Pervasive Computing and Communications (PerCom 03)*, 2003.

[85] Tian He, Chengdu Huang, Brian M. Blum, John A. Stankovic, and Tarek Abdelzaher. Range-free localization schemes for large scale sensor networks. In *Proceedings of the Ninth Annual International Conference on Mobile computing and networking (MobiCom'03)*, pages 81–95. ACM Press, 2003.

[86] Edith Pulido Herrera, Ricardo Quiros, and Hannes Kaufmann. Analysis of a Kalman Approach for a Pedestrian Positioning System in Indoor Environments. In *Euro-Par 2007 Parallel Processing*, volume 4641/2007 of *LNCS*, pages 931–940. Springer Berlin / Heidelberg, 2007.

[87] J. Hightower, R.Want, and G.Borriello. SpotON: An indoor 3D location sensing technology based on RF signal strength. Technical Report UW-CSE 00-02-02, University of Washington, Seattle, February 2000.

[88] Jeffrey Hightower. *The Location Stack*. PhD thesis, Department of Computer Science & Engineering, University of Washington, Seattle, WA, 2004.

[89] Jeffrey Hightower and Gaetano Borriello. A Survey and Taxonomy of Location Sensing Systems for Ubiquitous Computing. Technical report, University of Washington, Departement of Computer Science and Engineering, 2001.

[90] Jeffrey Hightower and Gaetano Borriello. Location Systems for Ubiquitous Computing. *IEEE Computer*, 34(8):57–66, 2001.

[91] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*. New York, NY: Springer-Verlag, Third Edition, 1992.

[92] Lingxuan Hu and David Evans. Localization for Mobile Sensor Networks. In *Proceedings of the Tenth Annual International Conference on Mobile Computing and Networking (MobiCom 2004)*, September 2004.

[93] Scott James and Boris Dragovic. Audio Location: Accurate Low-Cost Location Sensing. In *Proceedings of Pervasive 2005*, 2005.

[94] A. Janin, K.Zikan, D.Mizell, M.Banner, Sowizral, Jane Hsu, and Yung-Jen. A videometric head tracker for augmented reality applications. In *SPIE Telemanipulator and Telepresence Technologies*, 1994.

[95] Xiaodong Jiang, Nicholas Y. Chen, Jason I. Hong, Kevin Wang, Leila Takayama, and James A. Landay. Siren: Context-aware Computing for Firefighting. In *Proceedings of Pervasive 2004*, 2004.

[96] Xiaodong Jiang, Jason I. Hong, Leila A. Takayama, and James A. Landay. Ubiquitous computing for firefighters: field studies andprototypes of large displays for incident command. In *Proceedings of the SIGCHI conference on Humanfactors in computing systems(CHI '04)*, pages 679–686, New York, NY, USA, 2004. ACM Press.

[97] E.D. Kaplan. *Understanding GPS principles and applications*. Artech House, 2005.

[98] Katie Kelley. Firefighter Tracking. *Mission Critical Communications*, 22:47–54, 2007.

[99] Thomas King and Mikkel Baun Kjaergaard. ComPoScan: Adaptive Scanning for Efficient Concurrent Communications and Positioning with 802.11. In *Proceedings of the Sixth ACM International Conference on Mobile Systems, Applications, and Services (MobiSys 2008)*, 2008.

[100] Dana Kirsch and Thad Starner. The Locust Swarm: An environmentallypowered, networkless location and messaging system. In *In Proceedings of the First International Symposium on Wearable Computers, Boston, USA*, October 1997.

[101] Mikkel Baun Kjærgaard. A Taxonomy for Radio Location Fingerprinting. In *Proceedings of the Third International Symposium on Location and Context-Awareness (LoCA 2007)*, pages 139–156, 2007.

[102] Markus Klann. Playing with Fire: User-Centered Design of Wearable Computing for Emergency Response. In *First International Workshop on Mobile Information Technology for Emergency Response, Mobile Response 2007, Sankt Augustin, Germany, February 22-23, 2007*, volume 4458/2008, 2007.

[103] Markus Klann, Till Riedel, Hans Gellersen, Carl Fischer, Matt Oppenheim, Paul Lukowicz, Gerald Pirkl, Kai Kunze, Monty Beuster, Michael Beigl, Otto Visser, and Mirco Gerling. LifeNet: an Ad-hoc Sensor Network and Wearable System to

Provide Firefighters with Navigation Support. In *Adjunct Proc.eedings of Ubicomp 2007*, pages 124–127, 2007.

[104] Krysztof W. Kolodziej and Johan Hjelm. *Local Positioning Systems LBS Applications and Services*. Taylor and Francis, 2006.

[105] B. Krishnamachari. *Networking Wireless Sensors*. Cambridge University Press, 2005.

[106] Albert Krohn, Michael Beigl, Mike Hazas, Hans Gellersen, and Albrecht Schmidt. Using Fine-Grained Infrared Positioning to Support the Surface-Based Activities of Mobile Users . In *Proceedings of Fifth International Workshop on Smart Appliances and Wearable Computing (IWSAWC) 2005, Columbus, USA*, 2005.

[107] Albert Krohn, Michael Beigl, Mike Hazas, Hans-Werner Gellersen, and Albrecht Schmidt. Using Fine-Grained Infrared Positioning to Support the Surface-Based Activities of Mobile Users. In *Proceedings of Twenty-fifth IEEE International Conference on Distributed Computing Systems Workshops, 2005*, pages 463–468, 2005.

[108] J. Krumm, G.Cerma, and E.Horvitz. RightSPOT: A Novel Sense of Location for a Smart Personal Object. In *Proceedings of Ubicomp 2003*, pages 36–43, Seattle, WA, 2003.

[109] John Krumm, Steve Harris, Brian Meyers, Barry Brumitt, Michael Hale, and Steve Shafer. Multi-Camera Multi-Person Tracking for EasyLiving. In *Proceedings of the Third IEEE International Workshop on Visual Surveillance (VS'2000)*, pages 3–10. IEEE Computer Society, 2000.

[110] John Krumm and Eric Horvitz. LOCADIO: Inferring Motion and Location from Wi-Fi Signal Strengths. In *First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (Mobiquitous 2004)*, pages 4–13, August 2004.

[111] Vijay Kumar, Daniela Rus, and Sanjiv Singh. Robot and Sensor Networks for First Responders. *IEEE Pervasive Computing*, 3:24–33, 2004.

[112] Andrew M. Ladd, Kostas E. Bekris, Algis Rudys, Lydia E. Kavraki, Dan S. Wallach, and Guillaume Marceau. Robotics-based location sensing using wireless ethernet. In *Proceedings of the Eighth Annual International conference on Mobile computing and networking (MobiCom '02)*, pages 227–238. ACM Press, 2002.

[113] Andrew M. Ladd, Kostas E.Bekris, Algis Rudys, Guillaume Marceau, Lydia E.Kavraki, and Dan S. Wallach. Robotics based Location sensing using Wireless Ethernet. In *Proceedings of the Eigth ACM International Conference on Mobile Computing and Networking (MOBICOM)*.

[114] A. LaMarca, J.Hightower, I.Smith, and S.Consolvo. Selfmapping in 802.11 location systems. In *Proceedings of the Seventh International Conference on Ubiquitous Computing (Ubicomp 2005), Lecture Notes in Computer Science*, pages 87–104. Springer-Verlag, 2005.

[115] Anthony LaMarca, Yatin Chawathe, Sunny Consolvo, Jeffrey Hightower, Ian Smith, James Scott, Tim Sohn, James Howard, Jeff Hughes, Fred Potter, Jason Tabert, Pauline Powledge, Gaetano Borriello, and Bill Schilit. Place Lab: Device Positioning Using Radio Beacons in the Wild. In *Proceedings of Pervasive*, page pp. 116 133, Munich, Germany., 2005.

[116] Anthony LaMarca, Yatin Chawathe, Sunny Consolvo, Jeffrey Hightower, Ian Smith, James Scott, Tim Sohn, James Howard, Jeff Hughes, Fred Potter, Jason Tabert, Pauline Powledge, Gaetano Borriello, and Bill Schilit. Place Lab: Device Positioning Using Radio Beacons in the Wild. In *Proceedings of Pervasive'05*, May 2005.

[117] Koen Langendoen, Aline Baggio, and Otto Visser. Murphy loves Potatoes: Experiences from a Pilot Sensor Network Deployment in Precision Agriculture. In *Proceedings of Fourteenth International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS)*, April 2006.

[118] Koen Langendoen and Niels Reijers. Distributed localization in wireless sensor networks: a quantitative comparison. *Computer Networks*, 43(4):499–518, 2003.

[119] M. Langheinrich. Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems. In *Proceedings of Ubicomp 2001*, Atlanta, GA., 2001.

[120] R.B. Langley. Why is the GPS Signal So Complex. *GPS World Magazine*, 1(3):56–59, May/June 1990.

[121] Scott Lederer, Anind K. Dey, and Jennifer Mankoff. Everyday Privacy in Ubiquitous Computing Environments.

[122] J. Li, J.Jannotti, D.S.J.DeCouto, D.R.Karger, and R.Morris. A scalable location service for geographic adhoc routing. In *Proceedings of ACM Mobile Communications Conference*, 2000.

[123] Lin Liao, Donald J. Patterson, Dieter Fox, and Henry Kautz. Inferring High-Level Behavior from Low-Level Sensors. In *Proceedings of Fifth International Conference on Ubiquitous Computing, UBICOMP2003, Seattle, WA*, 2003.

[124] Tsung-Han Lin, Polly Huang, Hao hua Chu, and Chuang-Wen You. Energy-efficient Boundary Detection for RF-Based Localization Systems. *IEEE Transaction on Mobile Computing*, 7(12), December 2008.

[125] Ho lin Chang, Jr ben Tian, Tsung-Te Lai, Hao-Hua Chu, and Polly Huang. Spinning beacons for precise indoor localization. In *Proceedings of the Sixth ACM conference on Embedded network sensor systems(SenSys '08)*, pages 127–140, New York, NY, USA, 2008. ACM.

[126] Xiaotao Liu, Mark D. Corner, and Prashant Shenoy. Ferret: RFID Localization for Pervasive Multimedia. In *Proceedings of Ubiquitous Computing (Ubicomp'06')*, 2006.

[127] LogicaCMG. Making waves: RFID adoption in returnable packaging. `http://www.logicacmg.com/pdf/RFID.pdf`, 2004.

[128] Konrad Lorincz and Matt Welsh. MoteTrack: a robust, decentralized approach to RF-based location tracking. *Personal and Ubiquitous Computing*, 11(6):489–503, 2006.

[129] Paul Lukowicz, Andread Timm-Giel, Michael Lawo, and Otthein Herzog. WearIT@work: Toward Real-World Industrial Wearable Computing. *IEEE Pervasive Computing*, 6(4):8–13, December 2007.

[130] Anil Madhavapeddy and Alastair Tse. A Study of Bluetooth Propagation Using Accurate Indoor Location Mapping. In *Proceedings of Seventh International Conference on Ubiquitous Computing*, volume 3660 of *Lecture Notes on Computer Science*, pages 105–122. Springer, August 2005.

[131] Michael McCarthy, Paul Duff, Henk L. Muller, and Cliff Randell. Accessible Ultrasonic Positioning. *IEEE Pervasive Computing*, 5(4):86–93, 2006.

[132] Noel McKeegan. Tracking system promises faster help for avalanche victims. `http://www.gizmag.com/faster-help-for-avalanche-victims/8464/`, 2007.

[133] S. Meguerdichian, F.Koushanfar, M.Potkonjak, and M.B.Srivastava. Coverage problems in wireless adhoc sensor network. In *Proceedings of INFOCOM*, 2001.

[134] R. Mehra. On the identification of variances and apdaptive kalman filtering. *Transactions on Automatic Control*, 15(2):175–184, April 1970.

[135] L.E. Miller. RFID-Assisted Indoor Localization and Communication for First Responders. In *Proceedings of Tenth International Symposium on Advanced Radio Technologies (ISART 2006)*, 2006.

[136] J.J. More. The Levenberg-Marquardt Algorithm: Implementation and Theory. Lecture Notes in Mathematics, vol. 630, G. Watson, ed., Springer, 1978, pp. 105116, 1978.

[137] Kavitha Muthukrishnan and Mike Hazas. Position Estimation from UWB Pseudorange and Angle-of-Arrival: A Comparison of Non-linear Regression and Kalman Filtering. In Aaron Quigley Tanzeem Choudary and Thomas Strang, editors, *Proceedings of LoCA 2009: Fourth International Symposium on Location and Context Awareness, Tokyo, Japan*, volume 5561 of *Lecture Notes in Computer Science*, pages 222–239. Springer-Verlag Berlin Heidelberg, May 2009.

[138] Kavitha Muthukrishnan, Maria Lijding, and Paul Havinga. Towards Smart Surroundings: Enabling Techniques and Technologies for Localisa- tion. In Thomas Strang and Claudia Linnhof-Popien, editors, *Proceedings of the International Workshop on Location and Context-Awareness (LoCA 2005) at Pervasive 2005*, volume 3479 of *Lecture Notes in Computer Science*, pages 350–362. Springer-Verlang Berlin Heidelberg 2005, 2005.

[139] Kavitha Muthukrishnan, Maria Lijding, Nirvana Meratnia, and Paul Havinga. Sensing motion using spectral and spatial analysis of WLAN RSSI. In Gerrd Kortuem, Joe Finney, and Vasughi Sundramoorthy, editors, *Proceedings of the Second European Conference on Smart Sensing and Context (EuroSSC 2007)*, number ISBN 978-3-540-75695-8 in Lecture Notes in Computer Science, pages 62–76. Springer-Verlag Berlin Heidelberg, October 2007.

[140] Kavitha Muthukrishnan, Nirvana Meratnia, Maria Lijding, Georgi Koprinkov, and Paul Havinga. WLAN Location Sharing through a Privacy Observant Architecture. In *Proceedings of the First International Conference on Communication System Software and Middleware (COMSWARE)*, number ISBN 0-7803-9575-1. IEEE Communication Society Press, Los Alamitos, California, January 2006.

[141] Kavitha Muthukrishnan, Berend Jan van der Zwaag, and Paul Havinga. Inferring motion and location usin WLAN RSSI. In *Second ACM International workshop on Mobile Entity Localization and Tracking in GPS-less environments, Co-located with Ubicomp 2009, Orlando, USA (accepted for publication)*, 2009.

[142] Radhika Nagpal, Howard Shrobe, and Jonathan Bachrach. Organizing a Global Coordinate System from Local Information on an Ad Hoc Sensor Network. In *Proceedings of the Second International Workshop on Information Processing in Sensor Networks (IPSN '03)*, volume 2634 of *Lecture Notes in Computer Science*, April 2003.

[143] National Institute of Standards and Technology. RFID Tags To Assist In Tracking First Responders. ScienceDaily `http://www.sciencedaily.com/releases/2006/03/060330182126.htm`, March 2006. Accessed 2008.01.11.

[144] Rudy Negenborn. Robot Localization and Kalman filters On finding your position in a noisy world. Master's thesis, Utrecht University, 2003.

[145] Lionel M. Ni, Yunhao Liu, Yiu Cho Lau, and Abhishek P. Patil. LANDMARC: Indoor Location Sensing Using Active RFID. *Wireless Networks. Special Issue on Pervasive Computing and Communications*, 10(6):701–710, 2004.

[146] Dragos Niculescu and Badri Nath. DV based positioning in ad hoc networks. *Telecommunication Systems*, 22(1-4):267–280, 2003.

[147] Dragos Niculescu and Badri Nath. Position and Orientation in Ad hoc Networks. *Elsevier journal of Ad Hoc Networks*,

2(2):133–151, April 2004.

[148] M Ocana, L Bergasa, M Sotelo, J Nuevo, and R Flores. Indoor robot localization system using WiFi signal measure and minimizing calibration effort. In *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE 2005)*, volume 4, pages 1545–1550, 2005.

[149] Lauro Ojeda and Johann Borenstein. Non-GPS Navigation for Emergency Responders. In *2006 International Joint Topical Meeting: Sharing Solutions for Emergencies and Hazardous Environments*, pages 12–15, 2006.

[150] Robert J. Orr and Gregory D. Abowd. The smart floor: a mechanism for natural user identification and tracking. In *CHI '00 extended abstracts on Human factors in computing systems*, pages 275–276. ACM Press, 2000.

[151] A. Otsasson, A.Vrshavsky, A.Lamarca, and E.de Lara. Accurate GSM Localization. In *Proceedings of Ubicomp 2005*, Tokyo, Japan, 2005.

[152] Scott Pace, Gerald Frost, Irving Lachow, David Frelinger, Donna Fossum, Donald K. Wassem, and Monica Pinto. *The Global Positioning System*, chapter GPS history, chronology and budgets, pages 237–270. RAND Coorporation, 1995.

[153] Shwetak N. Patel, Khai N. Truong, and Gregory D. Abowd. PowerLine Positioning: A Practical Sub-Room-Level Indoor Location System for Domestic Use. In *Proceedings of UbiComp 2006*, 2006.

[154] Mihai Marin Perianu. *Collaborative Wireless Sensor Networks in Industrial and Business Processes*. PhD thesis, University of Twente, 2008.

[155] Ron Peterson and Daniela Rus. Interacting with a Sensor Network. In *Proceedings of IEEE International Conference on Robotics and Automation, 2004 (ICRA '04)*, pages 180–186, 2004.

[156] PhilipsBV. Pi-node Wireless motion tracking. `http://wockets.wikispaces.com/file/view/Pi-Node.pdf`.

[157] Pinpoint 3D positioning system. `http://www.pinpointco.com`.

[158] Kris Pister. 29 Palms Fixed/Mobile Experiment - Tracking vehicles with a UAV-delivered sensor network. `http://robotics.eecs.berkeley.edu/~pister/29Palms0103/`. Accessed 2008.01.03.

[159] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992.

[160] Nissanka B. Priyantha, Anit Chakraborty, and Hari Balakrishnan. The Cricket location-support system. In *Proceedings of the Sixth annual international conference on Mobile computing and networking (MobiCom '00)*, pages 32–43, New York, NY, USA, 2000. ACM Press.

[161] Doug Ramsey. UCSD Engineers, ER Physicians and County First Responders Test New Technologies in Disaster Drill. `http://ucsdnews.ucsd.edu/newsrel/general/emerprep06.asp`, August 2006. Accessed 2008.01.11.

[162] Cliff Randell and Henk Muller. Context awareness by analysing accelerometer data. In In Blair MacIntyre and editors Bob Iannucci, editors, *Proceedings of Fourth International Symposium on Wearable Computers*, pages 175–176, 2000.

[163] Cliff Randell and Henk Muller. Low cost indoor positioning system. In *Proceedings of Ubicomp 2001, Ubiquitous Computing*, pages 4248,, Atlanta, Georgia, USA, September 2001. ACM, Springer-Verlag.

[164] V. Renaudin, B.Merminod, and M.Kasser. Optimal Data Fusion for Pedestrian Navigation based on UWB and MEMS. In *Proceedings of PLANS 2008*, 2008.

[165] Valérie Renaudin, Okan Yalak, and Phillip Tomé. Hybridization of MEMS and Assisted GPS for Pedestrian Navigation. *Inside GNSS*, 2(1):34–42, January 2007.

[166] Valérie Renaudin, Okan Yalak, Phillip Tomé, and Bertrand Merminod. Indoor Navigation of Emergency Agents. *European Journal of Navigation*, 5(3):36–45, July 2007.

[167] C. Rizos, M.B.Higgins, and S.Hewitson. New global navigation satellite system developments and their impact on survey service providers and surveyors. In *Proceedings of Spatial Intelligence, Innovation and Praxis (SSC2005):The national binneal Conference of the Spatial Sciences Institute, Melbourne*, 2005.

[168] Z Sahinoglu, S Gezici, and I Guvenc. *Ultra-wideband Positioning Systems : Theoretical Limits, Ranging Algorithms and Protocols*. Cambridge University Press, 2008.

[169] Chris Savarese, Jan M. Rabaey, and Koen Langendoen. Robust Positioning Algorithms for Distributed Ad-Hoc Wireless Sensor Networks. In *Proceedings of the General Track: 2002 USENIX Annual Technical Conference*, pages 317–327. USENIX Association, 2002.

[170] Chris Savarese, J.M. Rabaey, and J. Beutel. Locationing in Distributed Ad-hoc Wireless Sensor Networks. In *Proceedings of ICASSP 2001*, 2001.

[171] Y. Shang and W.Ruml. Improved MDS-based localization. *In Proceedings of Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004)*, 4, 2004.

[172] J. Small, A.Smailagic, and D.P.Siewiorek. Determining user location for context aware computing through the use of wireless

LAN infrastructure, December 2000.

[173] A. Smith, H.Balakrishnan, M.Goraczko, and N.B.Priyantha. Tracking moving devices with the cricket location system. In *Proceedings of Second International Conference on Mobile Systems, Applications and Services (Mobisys 2004)*, June 2004.

[174] Timothy Sohn, Alex Varshavsky, Anthony LaMarca, Mike Y. Chen, Tanzeem Choudhury, Ian Smith, Sunny Consolvo, Jeffrey Hightower, William G. Griswold, and Eyal de Lara. Mobility Detection Using Everyday GSM Traces. In *Proceedings of the Eighth International Conference on Ubiquitous Computing. Irvine*, pages 212–224, September 2006.

[175] M.B. Srivastava, R.Muntz, and M.Potkonjak. Smart Kindergarden: Sensor-based Wireless Networks for Smart Developmental Problem-Solving Environments. In *Proceedings of ACM-IEEE Mobicom 2001*, 2001.

[176] Pete Steggles and Jay Cadman. A comparison of RF tag location products for real world applications. Technical report, Ubisense limited, March 2004.

[177] Daniel Steingart, Joel Wilson, Andrew Redfern, Paul Wright, Russell Romero, and Lloyd Lim. Augmented Cognition for Fire Emergency Response: An Iterative User Study. In *Proceedings of the First International Conference on Augmented Cognition*, 2005.

[178] TechWeb News. Google Acquires Mobile Social-Networking Company Dodgeball.com. `http://www.informationweek.com/news/global-cio/showArticle.jhtml?articleID=163101669`.

[179] TECO. The Particle Computer. `http://particle.teco.edu/`.

[180] S. Thrun. Robotic mapping: A survey. In Exploring Artificial Intelligence in the New Millenium. Morgan Kaufmann, 2002, 2002.

[181] S. Thrun, D.Fox, W.Burgard, and F.Dellaert. Robust Monte Carlo Localization for Mobile Robots. *Artificial Intelligence*, 128(1-2):99–141, 2000.

[182] Ubisense. Ubisense Technologies. http://www.ubisense.net/.

[183] Ubisense. Ubisense Location Engine Config Manual, 2007.

[184] A. Varshavsky, M.Chen.and E.de Lara, J.Froehlich, D.Haehnel, J.Hightower, A.LaMarca, F.Potter, T.Sohn, K.Tang, and I.Smith. Are GSM phones THE solution for localization? In *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications.=*, 2006.

[185] Versus. Versus Personnel Alert Badge. `http://ascc-inc.com/Healthcare/Versus.htm`.

[186] Roy Want, Andy Hopper, Veronica Falco, and Jonathan Gibbons. The active badge location system. *ACM Trans. Inf. Syst.*, 10(1):91–102, 1992.

[187] A. Ward. Calibration of a Location System, International Application No. PCT/GB2007/001425, Ubisense Limited, Nov 2007.

[188] A. Ward, A.Jones, and A.Hopper. A New Location Technique for the Active Office. *IEEE Personnel Communications*, 4:42–47, 1997.

[189] Andrew Martin Robert Ward. *Sensor driven Computing*. PhD thesis, Corpus Christi College ,University of Cambridge, 1998.

[190] Mark Weiser. The Computer for the twenty-first century. *Scientific American*, 265:94–104, 1991.

[191] Eric Weisstein. Spearman Rank Correlation Coefficient. From MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/.

[192] Greg Welch and Gary Bishop. SCAAT: Incremental Tracking with Incomplete Information. *Computer Graphics*, 31:333–344, 1997.

[193] Greg Welch and Gary Bishop. An Introduction to Kalman filter. Technical Report Tech. Rep. 95-041, Dept of Computer science, University of North Carolina, 2006 (updated).

[194] Greg Welch, Gary Bishop, Leandra Vicci, Stephen Brumback, Kurtis Keller, and D'nardo Colucci. The HiBall Tracker: high-performance wide-area tracking for virtual and augmented environments. In *Proceedings of the ACM symposium on Virtual reality software and technology (VRST '99)*. ACM, 1999.

[195] Greg Welch and Eric Foxlin. Motion Tracking: No Silver Bullet, but a Respectable Arsenal. *IEEE Comput. Graph. Appl.*, 22(6):24–38, 2002.

[196] Martyn Welch. Techniques for angle estimation in ultrasonic positioning systems. Master's thesis, Lancaster University, November 2005.

[197] Alan F. Westin. *Privacy and Freedom*. Atheneum, New York NY, 1967.

[198] WhereNet. http://www.wherenet.com/.

[199] XSens. XSens website. `http://www.xsens.com`. Accessed 2008.01.09.

[200] S.H. Yang and P.Frederick. SafetyNET - a wireless sensor network for fire protection and emergency responses. *Measurement and Control*, 39:218–219, 2006.

[201] Shun-Yuan Yeh, Keng-Hao Chang, Chon-In Wu, Hao-Hua Chu, and Jane Yung-Jen Hsu. GETA sandals: a footstep location tracking system. *Personal and Ubiquitous Computing*, 11:451–463, 2007.

[202] Moustafa Youssef and Ashok Agrawala. The Horus WLAN location determination system. In *Proceedings of the Third International Conference on Mobile systems, applications, and services (MobiSys '05)*, pages 205–218, New York, NY, USA, 2005. ACM Press.

[203] Moustafa Youssef, Ashok Agrawala, and Udaya Shankar. WLAN Location Determination via Clustering and Probability Distributions. In *Proceedings of IEEE PerCom 2003 (PerCom03)*, 2003.

[204] Ying Zhang, Lee Ackerson, David Duff, Craig Eldershaw, and Mark Yim. STAM: a system for tracking and mapping in real environments. *IEEE Wireless Communications*, 11:87–96, December 2004.

Publications

1. Kavitha Muthukrishnan, Berend Jan van der Zwaag and Paul Havinga, "Inferring motion and location using WLAN RSSI", *In Proceedings of the Second ACM International workshop on Mobile Entity Localization and Tracking in GPS-less environments, Co-located with Ubicomp 2009*, 30 Sep - 3 October 2009, Orlando, USA (to appear).

2. Kavitha Muthukrishnan and Mike Hazas, "Position Estimation from UWB Pseudorange and Angle-of-Arrival: A Comparison of Non-linear Regression and Kalman Filtering", *In Proceedings of Fourth International Symposium on Location and Context Awareness (LoCA 2009)*, Lecture Notes in Computer Science, volume 5561, pp. 222-239, Tokyo, Japan, May 2009.

3. Carl Fischer, Kavitha Muthukrishnan, Mike Hazas and Hans Gellersen, "Ultrasound-aided Pedestrian Dead Reckoning for Indoor Navigation", *In Proceedings of the First ACM International workshop on Mobile Entity Localization and Tracking in GPS-less environments, Co-located with Mobicom 2008*, 15-19 September 2008, San Francisco, USA, pp. 31-36, ACM ISBN 978-1-60558-189-7.

4. Kavitha Muthukrishnan, Maria Lijding, Nirvana Meratnia, Paul Havinga, "Sensing motion using spectral and spatial analysis of WLAN RSSI", *In Proceedings of the Second European Conference on Smart Sensing and Context (EuroSSC 2007)*, 23-25 October 2007, Lake District, UK, pp. 62-76, ISBN 978-3-540-75695-8.

5. Kavitha Muthukrishnan, Nirvana Meratnia, Maria Lijding, Georgi Koprinkov and Paul Havinga, "WLAN Location Sharing through a Privacy Observant Architecture", *In Proceedings of the First International Conference on Communication System Software and Middleware (COMSWARE)*, published by IEEE Communication Society Press, Los Alamitos, California, January 2006, ISBN 0-7803-9575-1.

6. Kavitha Muthukrishnan, Nirvana Meratnia, Maria Lijding, Georgi Koprinkov and Paul Havinga, "Demonstrating FLAVOUR: Friendly Location-aware conference Assistant with priVacy Observant architectURe", *In Demonstration Session Proceedings of the Third International Conference on Service Oriented Computing (ICSOC'05)*, 12-15 December 2005, Amsterdam, The Netherlands pp. 13-16, Vita Lab, ISBN 3-540-25896-5.

7. Kavitha Muthukrishnan, Nirvana Meratnia, Maria Lijding, Georgi Koprinkov and Paul Havinga, "SVGOpen Conference Guide: An overview", *In Proceedings of the Fourth Annual Conference on Scalable Vector Graphics (SVG'05)*, 15-18 August 2005, Enschede, The Netherlands.

8. Kavitha Muthukrishnan, Maria Lijding and Paul Havinga, "Towards Smart Surroundings: Enabling Techniques and Technologies for Localisation", *In Proceedings of the International Workshop on Location and Context-Awareness (LoCA 2005) co-located with Pervasive 2005*, pp. 350-362, May 2005, ISBN 3-540-25896-5.