Latitude, Longitude, and Beyond

Mining Mobile Objects' Behavior

Mitra Baratchi

Latitude, Longitude, and Beyond

Mining Mobile Objects' Behavior

Graduation committee:

Promotor:	Prof.dr. P.J.M. Havinga
Promotor:	Prof.dr. A.K. Skidmore
Co-promotor	Dr. N. Meratnia
Members:	
Prof. dr. M.R. van Steen	University of Twente
Prof. dr. P.M.G. Apers	University of Twente
Prof. dr. K.G. Langendoen	Delft University of Technology
Prof. dr. J.J. Lukkien	Eindhoven University of Technology
Dr. A.G. Toxopeus	University of Twente, ITC
Dr. L.P.J.J. Noldus	Noldus Information Technology BV

CTIT Ph.D-thesis Series No. 15-366.

Center for Telematics and Information Technology

University of Twente

P.O. Box 217, NL – 7500 AE Enschede

ISSN 1381-3617

ISBN 978-90-365-3892-3

Publisher: Gildeprint, Enschede

Cover design: Sadaf Nadimi, sadaf.nadimi@gmail.com

LATITUDE, LONGITUDE, AND BEYOND

MINING MOBILE OBJECTS' BEHAVIOR

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 the 24th of June 2015 at 12:45

by

Mitra Baratchi

born on the 22nd of September 1985
in Tehran, Iran

This dissertation has been approved by:

Promotor:	Prof. dr. ing. P.J.M. Havinga
Promotor:	Prof. dr. ir. A.K. Skidmore
Co-promotor:	Dr. ir. N. Meratnia

Acknowledgments

I believe I have been extremely lucky and blessed by meeting so many amazing people in my life who inspired me and made me the person I am now. I am grateful to them all, even though it is impossible to name everyone.

Herewith, I express my deepest gratitude to some of the people who helped me the most within the past four years:

My promotors, prof. Paul Havinga and prof. Andrew Skidmore, for giving me the opportunity of doing this fabulous research, the freedom to choose my own path within it, and their prompt directions along the way. My supervisor, dr. Nirvana Meratnia, for guiding me on a daily basis, revising all my last minute papers, and for her understanding all the issues and situations, which were not always work-related. I am also thanking dr. Bert Toxopeus for being involved in my research from the beginning and his input to my research and papers.

All my committee members, prof. Maarten van Steen, prof. Peter Apers, prof. Koen Langendoen, prof. Johan Lukkien, and dr. Lucas Noldus for accepting to be in my committee and their valuable comments.

My most inspiring teachers in life, my first maths teacher, my mom, my other maths teacher, Taravat Moshtagh, and my Dutch teacher, Natasja van Dulmen. If I pursue a teaching career, I will definitely take you three as role models.

My previous colleagues at Pervasive Systems, especially my office mates, Yang Zhang, Arta Dilo, and Okan Turkes and, my friends who participated in data collection, Zahra Taghikhaki, Alireza Masoum, Pooria Zand, Guohua Yang, and Okan Turkes. I have a special thank you for Berend Jan Van der Zwaag for editing my longest paper and Kyle Zhang for helping me with the GPS data logger.

My colleagues at Saxion, especially my supervisor, dr. Wouter Teeuw, for encouraging me to finish my PhD, and Peter Ebben for translating the abstract of my thesis.

My friends from back home and here in Enschede, Mozhde Gholibeigi, Morteza Karimzadeh, Neda Mostafa, Mohamad Mozaffari, Mahroo Zandrahimi, Sina Behfard, Mohamadreza Khelghati, Elahe Yeganegi, Farhad Vishkai, Mojtaba Farmanbar, Niloofar Tahmasebi, and Mina Shahi. I also thank all the other members of the IRNUT community for celebrating all the important Persian events together.

My paranymphs, Alireza Rohani and Sadaf Nadimi, for helping me during the past few months. Another thank you to Sadaf for designing the illustrative cover.

All my teachers and friends form Dutch courses at UT, for all the multicultural talks and fun activities we did together, as well as learning Dutch.

My parents in-law, my uncle in law, Shahin, and his wife, Liesbet, for making me feel at home from the first day I came to the Netherlands.

My parents, for always being supportive of all the decisions I made in my life, and letting me follow my dreams. My sister, Sara, and brother in law, Khashayar, for being my guide in every other aspect. My adorable nephew, Arash, for bringing so much joy and happiness with himself. I also cherish the memory of my brother, Hamidreza, who is not with us anymore.

My husband and my best friend, Siavash, for his love, support, and presence, which give me the courage to move forward and have no fear of challenges.

Mitra Baratchi

Enschede, June 2015

Abstract

Rapid advancements in Micro-Electro-Mechanical Systems (MEMS), and wireless communications, have resulted in a surge in data generation. Mobility data is one of the various forms of data, which are ubiquitously collected by different location sensing devices. Extensive knowledge about the behavior of humans and wildlife is buried in raw mobility data. This knowledge can be used for realizing numerous viable applications, ranging from wildlife movement analysis, to various location-based recommendation systems, urban planning, and disaster relief.

With respect to what mentioned above, in this thesis, we mainly focus on providing data analytics for understanding the behavior and interaction of mobile entities (humans and animals). To this end, the main research question to be addressed is:

How can behaviors and interactions of mobile entities be determined from mobility data acquired by (mobile) wireless sensor nodes in an accurate and efficient manner?

To answer the above-mentioned question, both application requirements and technological constraints need to be considered. On the one hand, applications requirements call for accurate data analytics to uncover hidden information about individual behavior and social interaction of mobile entities, and to deal with the uncertainties in mobility data. Technological constraints, on the other hand, require these data analytics to be efficient in terms of their energy consumption and to have low memory footprint, and processing complexity.

The contributions of this thesis are:

- **Mining periodic behavior from mobility data:** Periodic behaviors are prevalent for both humans and wildlife. We propose a technique for identifying periodic behaviors and extracting periodic patterns from streaming mobility data.
- **Modeling mobility data:** A general movement model can be used to identify frequent patterns in the mobility data using the higher-level semantic they represent. We model trajectories both deterministically and probabilistically to relate them to the paths and stay-points they represent. Our deterministic approach uses collective knowledge of trajectories (on the move) to relate trajectories to the path taken by mobile entities. Our probabilistic approach, models trajectories (on stay-points and on the move) using state-space modeling techniques.

- **Mining social ties only from mobility data:** We study the possibility of extracting social context from mobility data. For this purpose, we propose two information theoretic indicators to measure the correlation between visits to different places based on the purpose of visit.
- **Model-based trajectory compression and adaptive sampling:** We propose two techniques to use the patterns discovered by trajectory modeling to reduce data redundancy and uncertainty. Thereby, we increase the lifetime of the location acquisition devices. Our first technique is a light online model-based trajectory compression algorithm for decreasing the number of transmitted samples. Our second approach is a model-based adaptive sampling algorithm, which increases the lifetime of the location acquisition devices by reducing both the number of samples acquired and transmitted.

The techniques developed in the thesis were evaluated using five different mobility datasets collected from both wildlife and humans. These datasets are: i) small-scale dataset collected by a wireless sensor node carried by people, ii) small-scale dataset collected from capricorns, iii) large-scale Geolife dataset from Microsoft research, iv) large-scale Mobile Data Challenge dataset from Nokia, and v) a synthetic dataset produced with a movement test sequence generator.

Samenvatting

Snelle vooruitgang in de ontwikkeling van micro-elektromechanische systemen (MEMS) en draadloze communicatie heeft geresulteerd in een enorme toename in gegenereerde data. Mobiliteitsdata is een van de vormen van data die overal verzameld wordt, gebruikmakend van verschillende locatiebewuste apparaten. Uitgebreide kennis over het gedrag van mensen en wilde dieren is verborgen in deze ruwe mobiliteitsdata. Deze kennis kan gebruikt worden voor het realiseren van talrijke zinvolle applicaties, variërend van het analyseren van de beweging van dieren in het wild tot verschillende locatie-gebaseerde *recommendation systems*, stadsplanning en hulpverlening bij rampen.

In het licht van het bovenstaande, richten we ons in dit proefschrift vooral op *data analytics* om het gedrag en de interactie van mobiele entiteiten (mensen en dieren) te begrijpen. De hoofdvraag die in het proefschrift beantwoord wordt, is:

Hoe kan het gedrag en de interactie van mobiele entiteiten op een accurate en efficiënte wijze bepaald worden op basis van mobiliteitsdata verzameld door (mobiele) draadloze sensoren?

Om bovenstaande vraag te beantwoorden moet rekening gehouden worden met zowel de eisen aan de applicaties als de technologische beperkingen. Aan de ene kant vragen applicaties om accurate *data analytics* om verborgen informatie over individueel gedrag en sociale interactie van mobiele entiteiten zichtbaar te maken en moeten de algoritmes om kunnen gaan met onzekerheden in mobiliteitsdata. Technologische beperkingen, aan de andere kant, vereisen dat deze algoritmes efficiënt zijn in termen van hun energieverbruik, een laag geheugengebruik en lage (reken)complexiteit hebben.

De bijdragen van dit proefschrift zijn:

- **Afleiden van periodiek repeterend gedrag uit mobiliteitsdata:** Periodiek repeterend gedrag overheerst bij zowel mensen als wilde dieren. We presenteren een techniek om periodiek repeterend gedrag te identificeren en periodiek repeterende patronen af te leiden uit streaming mobiliteitsdata.
- **Modelleren van mobiliteitsdata:** Een algemeen model van verplaatsingen kan gebruikt worden om veel voorkomende patronen in de mobiliteitsdata te identificeren, gebruikmakend van de hogere niveau semantiek die deze patronen representeren. We modelleren afgelegde trajecten zowel deterministisch als probabilistisch om ze te relateren aan de routes en

verblijfplaatsen die ze representeren. Onze deterministische aanpak gebruikt collectieve kennis van afgelegde trajecten (tijdens verplaatsingen) om deze trajecten te relateren aan de door de mobiele entiteiten afgelegde route. Onze probabilistische aanpak modelleert afgelegde trajecten (zowel op verblijfplaatsen als daartussen) gebruikmakend van state-space modelleertechnieken.

- **Afleiden van sociale relaties op basis van alleen mobiliteitsdata:** We bestuderen de mogelijkheid om de sociale context af te leiden uit mobiliteitsdata. Voor dit doel presenteren we twee informatietheoretische indicatoren om de correlatie tussen bezoeken aan verschillende locaties te meten, gebaseerd op het doel van het bezoek.
- **Modelgebaseerde traject-compressie en adaptieve dataverzameling:** We presenteren twee technieken om de patronen die ontdekt zijn op basis van het modelleren van de afgelegde trajecten te gebruiken om redundantie in de data en onzekerheid te verminderen. Hiermee vergroten we de levensduur van de locatiebepalingsapparatuur. Onze eerste techniek is een licht, modelgebaseerd traject-compressie algoritme voor het online reduceren van de hoeveelheid metingen die verstuurd worden. Onze tweede aanpak is een modelgebaseerd adaptief samplingsalgoritme dat de levensduur van de locatiebepalingsapparatuur vergroot door zowel het aantal metingen dat verricht wordt als het aantal metingen dat verstuurd wordt, te reduceren.

De in dit proefschrift ontwikkelde technieken zijn geëvalueerd, gebruikmakend van vijf verschillende mobiliteitsdatasets verzameld bij zowel wilde dieren als mensen. Deze datasets zijn: i) kleinschalige dataset verzameld door een draadloze sensor, gedragen door mensen, ii) kleinschalige dataset verzameld bij steenbokken, iii) grootschalige *Geolife* dataset van Microsoft research, iv) grootschalige *Mobile Data Challenge* dataset van Nokia, en v) een synthetische dataset geproduceerd met een *movement test sequence* generator.

Contents

Chapter 1

Introduction

1.1 Introduction

With the emergence of sensor technologies and advances in mobile and wireless communications, more and more data are being generated at all times. The surge in the rate of data generation is so rapid that, supposedly more than 90% of the data in the world was produced only in the past two years [1]. This has led to the birth of the new era of "Big Data" which, in simplest terms, refers to loads of data, which are far more than what a single computer can handle. Extensive knowledge is hidden in such abundance of data waiting to be revealed.

Spatio-temporal mobility data acquired through location aware technologies are only one of the various forms of data, which are continuously increasing in size. Investments on the effort and price spent on collecting mobility data (both in terms of knowledge acquisition, and technology development) are only returned when the data is effectively processed to realize a viable application. Therefore, now that data sensing and collection is easily possible, the need for techniques, which make sense of such data, is more than ever evident.

Any application that uses mobility data collected from a mobile entity requires answering various questions about it. At the same time, the technology limits the capability of current data analysis solutions by introducing its own challenges. Thereby, any solution for analyzing mobility data should meet certain requirements. In what follows, we further elaborate this problem by naming few applications, technological constraints, and the requirements imposed by the two.

1.2 Applications

Analysis and mining mobility data provided through sensing has provoked various applications. The following are only a handful of the possible environmental and civil applications:

- **Wildlife monitoring:** Possibility of collecting mobility data from wildlife has opened new avenues for ecologists interested in wildlife movement analysis. Data collected from animals can explain numerous phenomena in the ecological domain such as their resource selection [2], foraging [3], predation [4], and intersexual relationships [5].

- **Location based recommendation systems:** Thanks to mobility data analysis, various services can be provided for pedestrians and cars. For instance, information extracted from such data can realize location based recommendations (e.g. routes, venues) which match the users' preferences and their current location [6].
- **Environmental monitoring:** Recently, through the crowd participation in data collection (participatory sensing) [7], rich mobility datasets are formed. These datasets can be used for solving various environmental issues. For instance, researchers have used such data for visualizing city-wide spatio-temporal noise and pollution maps [8].
- **Urban planning:** Mobility datasets collected from people provide extensive knowledge about urban resource usage. This knowledge can be used as part of planning for future urban growth [9].
- **Disaster relief:** Movement analysis of people during disaster can help in disaster relief and management during severe catastrophes which involve large population movement [10].

1.2.1 Technological constraints

Mobility data needed for realizing the above-mentioned applications can be provided through various means. Human related data are ubiquitously collected through location aware devices such as mobile phones, PDAs, Wi-Fi networks, and location check-ins in social networking websites. During the past decade there has also been a growing interest in the use of mobile wireless sensor networks especially for wildlife monitoring. These networks are composed of autonomous mobile nodes for collecting data of interest [11].

The technological restrictions of the above-mentioned technologies impose numerous challenges on data analysis systems. Some of these challenges are:

- **Resource constraints:** Constraints in mobile sensing technologies are of different natures, mainly raised by cost and size restrictions. These restrictions have led to scarcity of energy, memory, computational capability, and communicational range and bandwidth.
- **Data uncertainty:** Uncertainties [12] are inherent in mobility data collected by mobile sensing devices. These uncertainties are in form of discrete sampling and missing samples. Discrete nature of sampling is the source of uncertainty between two consequent samples. The level of uncertainty in mobility data is affected by the frequency by which position samples are acquired [12]. In real world applications, there are also missing samples. This implies that data is unavailable due to hardware failure or transmission error, among other reasons.

- **Inaccuracy:** Mobility data should capture the physical location of the mobile entity. Oftentimes, inaccuracies are introduced to the estimated physical location in form of measurement error and noise. Measurement error is a minor deviation from the correct value. Noise, in this context, is a measurement, which does not make sense considering the maximum speed of the mobile entity. Such inaccuracies are also challenges that need to be tackled [12].
- **Lack of ground truth/context:** A great challenge which scientists face when analyzing mobility data is lack of available datasets with relevant ground-truth. Without available ground truth, it is very difficult to prove performance of any algorithm that derives context. At the same time, collecting a sufficiently large dataset, which is labeled with relevant ground truth, is energy and time consuming.

1.2.2 Requirements

Both of the formerly mentioned applications and challenges call for specific requirements for any data analysis system that is devised for analysis of mobility data:

- **Pattern mining and data analysis techniques:** All of these applications pose questions about the behavior of individuals and groups such as: “Where are the regions of interest?”, “What are the frequent paths?”, and “What are the social ties between individuals?”. Identifying the right method, which can correctly answer the above-mentioned questions, is one of the basic requirements of the mobility data analysis systems.
- **Knowledge discovery:** Multiple factors affect the behavior of mobile entities. Habits, social interactions, special events, and changes can potentially form patterns in the mobility data. Therefore, when it comes to explaining the behavior of a mobile entity, there is no single pattern to look for. Discovering unknown patterns present in the data without availability of a-priori knowledge is a difficult task. While intuition can help in making hypothesis about existence of patterns in the data collected from humans, there is less assistance from such insights when studying wildlife.
- **Streaming data analysis:** Various applications mentioned before require instant decision-making based on data. Mobility data collected from (mobile) wireless sensor nodes are in form of streams of $(Lat, Long, Alt, t)$ coordinates being generated continuously. Formerly, the collected data were analyzed in an offline mode. Nowadays, however, such form of analysis neither satisfies the requirements of applications in terms of responsiveness, nor is it possible considering the resource constrains of mobile devices.

- **Efficient means of data collection and processing:** The scarcities of mobile wireless sensor nodes in terms of memory and computational resources, as mentioned before, calls for efficiency. Therefore, usage of memory and processing resources during sampling, processing, and transmission should be efficient.

1.3 Research objectives

The main focus of this thesis is on providing data analysis mechanisms for understanding the behavior and interactions of mobile entities. To this end, the general research question to be addressed in this thesis is:

How can behaviors and interactions of mobile entities be determined from mobility data acquired by (mobile) wireless sensor nodes in an accurate and efficient manner?

With respect to the requirements mentioned in Section 1.2.2, there are two key factors to be considered in order to answer the above-mentioned question. The first, and most important aspect is imposed by application requirements, which are *data analysis solutions* that uncover hidden patterns from mobility data acquired by location aware technologies. The second aspect imposed by technological constraints is *efficient data collection*. Efficiency here relates to feasibility of collecting data in a timely manner, considering the resource scarcity, and in a way that important information regarding the patterns and behaviors are still captured.

1.3.1 Research questions

Considering the need for (i) application related data analysis solutions for understanding the data and (ii) technological infrastructure for data collection, we derive the three specific research questions (RQ 1-3) mentioned below:

Application related data analysis solutions for understanding the data:

- ***(RQ. 1)**: Can the acquired mobility data from a mobile entity be efficiently interpreted to provide knowledge about the individual behavior of mobile entities?*
- ***(RQ. 2)**: Can the acquired mobility data from mobile entities be interpreted to provide knowledge about their interaction?*

Technological infrastructure for data collection:

- ***(RQ. 3)**: Can the mobility data be collected efficiently in terms of resources and the required data frequency?*

1.3.2 Hypotheses

In order to answer the first research question (RQ. 1), we start with the hypothesis that it is possible to model the behavior of individual mobile entities in terms of the *periodic, and frequent* patterns in mobility data. We address (RQ. 2) by making the hypothesis that social context and social ties can form similar mobility patterns. Concerning the last research question (RQ. 3), we hypothesize that context-based long-term redundancies in mobility data (periodic or frequent) can be used to reduce the overhead caused by both sending and sampling data.

1.4 Thesis contribution

With respect to what we mentioned in the previous section, the contributions of this thesis can be summarized as below:

- **A review of technological solutions for collecting spatio-temporal data from mobile entities (Chapter 2):** The first and foremost question for data analysis is associated with the type of data to analyze. Therefore, the choice of proper technology, which is inline with the application needs, and the available resources, is of high importance. For this purpose, we investigate and review various technological solutions, which could be used for collecting spatio-temporal data from mobile entities. We provide an overview of different sensing technologies, classify them, and review their capabilities.
- **Efficient and accurate extraction of periodic patterns from streaming data (Chapter 3):** Periodic patterns are prevalent in both human's and animal's behavior. Although the problem of mining periodic patterns has been addressed before, there was no previous research on extracting patterns form streaming data. To this end, we propose a periodicity detection technique for streaming mobility data, which can extract periodic patterns, having low memory and processing footprints.
- **Efficient and accurate modeling of trajectory dynamics (Chapter 4):** A general movement model can be used to identify frequent patterns in mobility data. We propose using both deterministic and probabilistic modeling to capture the short-term dependencies in the mobility data. Our deterministic approach is a hierarchical grid-based clustering algorithm that utilizes the semantic and spatial data to find the frequently visited paths by mobile entities. This technique can accurately find frequent patterns by using the collective knowledge of trajectories.

  Our second approach is a probabilistic generative model, based on hierarchal hidden semi-Markov model (HHSMM), which can capture both frequent, and rare mobility patterns of mobile entities. As will be shown, this technique can generally model trajectories such that new patterns are better discovered without any presumptions.

- **Extracting social context from mobility data (Chapter 5):** Social context can be discovered from mobility patterns. The previous research in this area, normally use other types of data which are rich in social context (such as phone calls, and message posts on social networks). In contrast, we study the possibility of extracting social context solely from mobility data. For this purpose, we propose two information theoretic indicators to measure the correlation between visits to places based on the purpose of visit.
- **Energy efficient collection of mobility data (Chapter 6):** Data acquired from location aware devices are conforming to a large amount of spatio-temporal correlations, which make them extremely redundant. We propose two techniques to use the previously found patterns in the data to reduce such redundancies. Firstly, we use the patterns present in data for compressing trajectories. Secondly, we show how it is possible to use these redundancies in order to decrease the number samples acquired by mobility data acquisition devices. Using these two techniques, we increase the lifetime of the mobile sensor node by decreasing the resources required for sensing and transmission of data.
- **Experiments with different datasets (Chapters 3-6):** One of our contributions in this thesis is extensive experiment with different mobility datasets. The datasets we used in this thesis are versatile in terms of their scale and the amount of uncertainty they contain (sparseness and noise). Relevant to each of the contributions mentioned before, we have chosen datasets, which can convey information about individual or social behavior of mobile entities. At the end of each chapter, we have case studies on two specific datasets. More details on these datasets are provided in Section 1.5.

1.5 Datasets for experimental validations

We have chosen the following datasets for our experiments in different chapters of this thesis:

- **Dataset 1 - *PS dataset:*** This dataset is collected with custom designed sensor nodes shown in Figure 1-1. This sensor node houses different sensors such as a GPS sensor, light sensor, accelerometer and temperature sensor. Data collected by sensors is logged in a flash memory. For the studies performed in this thesis we used the GPS data sampled every 60 seconds. Between June-September 2012, 6 members of the Pervasive Systems research group at the University of Twente have carried these nodes. Data collected by these members is available for a period between three weeks and 109 days. The longer-term analysis performed in Chapters 3-6 was performed using the data collected by one of the candidates for 109 days. In Chapter 3, the data collected by the whole group, in a period of 21 days, is used.

Figure 1-1 The custom designed wireless sensor node used to collect PS dataset

- **Dataset 2 – *ITC Capricorn dataset* [13]:** This dataset is composed of GPS data collected by data loggers attached to three capricorns. Data from loggers were downloaded using UHF handheld devices in November 2011. The Cretan Capricorn (Capra aegagrus-cretica) lives in the White Mountains and is endemic for Crete. Due to increasing livestock populations (goats) their population is threatened. As the species is difficult to locate very little is known about their habitat use in different seasons. Since mid-July of 2011 one male and two female capricorns have been equipped with GPS collars. By deploying animal collars equipped with GPS, precise spatio-temporal data is provided in fixed time intervals [38]. The sampling frequency in this dataset is lower than the other datasets and only 16 GPS-samples where acquired per day. These samples have been acquired in the morning and in the afternoon based on the daily behavior of the Capricorn [14].
- **Dataset 3 - *Geolife dataset* [15-18]:** This dataset is collected during the GeoLife project organized by Microsoft Research Asia over a period of three years between April 2007 and August 2012. GPS samples collected by 182 users were recorded using both GPS loggers and GPS-phones. For the majority of users (91 percent) the sampling frequency is as high as reporting a sample every (1~5) seconds. The users have recorded mobility data during various activities and habits of their outdoor daily life in this dataset.
- **Dataset 4 - *Nokia Challenge dataset* [19, 20]:** This dataset was collected through an initiative by Nokia Research Center Lausanne and its Swiss academic partners Idiap and EPFL between January 2009 and March 2011. The aim of this data collection has been, generating innovation in smartphone-based research. This dataset consists of longitudinal smart-phone based data collected by over 200 volunteers in the Lake Geneva region using Nokia N95 phones and a client-server architecture that made the data collection invisible

to the participants. Other than mobility data (GPS, WLAN), this dataset also contains various other forms of data such as data types related to motion (accelerometer), proximity (Bluetooth), communication (phone call and SMS logs), multimedia (camera, media player), and application usage (user-downloaded applications in addition to system ones) and audio environment (optional) were recorded. Thereby, it not only encloses information about mobility behavior but also about social interactions.

- **Dataset 5 - *Synthetic dataset:*** This dataset is an artificial dataset produced with a movement generator. For certain experiments in Chapter 3 and 4, we needed to show robustness of the proposed algorithm in presence of noise and missing samples. Since these features were not always consistently present in the above datasets, we used this synthetic dataset to show how certain level of noise and missing samples affect the performance of the proposed techniques.

The Long-term goal of this research is using the mobile nodes used for collecting Dataset 1 for collecting mobility data from wildlife. With this respect, we base our data analysis on Dataset 1 and Dataset 2. In different chapters, with respect to the specific aims of that chapter we use other datasets as well.

Tables 1-1 and 1-2 summarize dataset characteristics and dataset usage per chapter.

		Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
Chapter 3		✓	✓			✓
Chapter 4	**4.1**	✓	✓			
	4.2	✓	✓	✓		✓
Chapter 5		✓	✓		✓	
Chapter 6		✓	✓			

Table 1-1 Datasets used per chapter

	Number of cases	Duration	Sampling interval	Data	Mobility data format
Dataset 1	6	2011.7-2011.10	60 seconds	GPS, Accelerometer, Temperature, light	Raw GPS data
Dataset 2	3	2011.7-2011.11	16 samples/day	GPS	Raw GPS data
Dataset 3	182	2007.4-2012.8	1-5 second	GPS	Raw GPS data
Dataset 4	200	2009.1-2011.3	10 seconds	GPS, Phone/Sms logs, app usage, audio environment	Raw GPS Stay-points
Dataset 5	1	-	60 minutes	GPS	Raw GPS

Table 1-2 Summary of datasets

1.6 Thesis organization

The organization of the thesis and the relationship among different chapters are illustrated in Figure 1-2. In Chapter 2, we review the state of the art technologies that can be used for collecting mobility data from humans and animals. In Chapter 3, the problem of mining periodic patterns from mobility data is discussed. Trajectory modeling is the subject of Chapter 4. Extraction of social interaction between mobile entities is the topic presented in Chapter 5. In Chapter 6, we tackle the problem of trajectory compression and energy efficient sensing of mobility data. A number of remarks presented in Chapter 7 conclude this thesis.

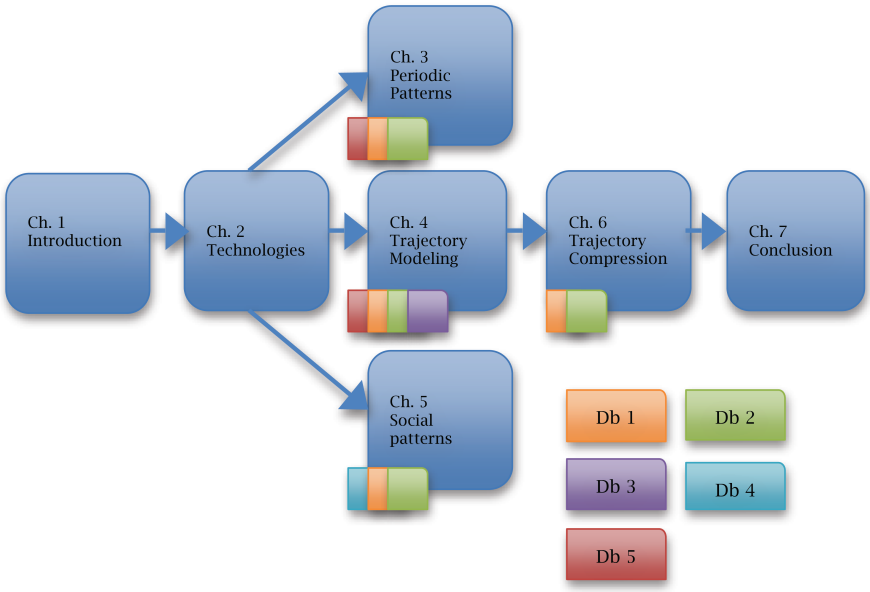


Figure 1-2 Relationship between chapters and datasets use

Chapter 2

Technological solutions for collecting spatio-temporal data from mobile entities[1]

To realize data mining applications, scientists rely on data collection technologies, which can provide spatio-temporal data for understanding movement paradigms. Recently, wireless sensor networks have offered new opportunities for data collection from remote places through their multi-hop communication and collaborative capability of nodes. Several technologies can be used in such networks for collecting spatio-temporal data from mobile objects. In this chapter, we investigate and review technological solutions, which can be used for collecting data for wildlife monitoring and human sensing applications. Our aim is to provide an overview of different sensing technologies and their capabilities in terms of the data they provide for modeling mobility behavior of mobile entities. First, we classify the sensors based on the movement-modeling approach they are used for, and then review data types that can be acquired using each sensor. Finally, we compare these sensing technologies in terms of their limitations, advantages, and the data they can provide.

2.1 Classification of technologies for collecting spatio-temporal data

Modeling movement using spatio-temporal data is generally performed using two approaches, i.e., (i) the Lagrangian approach and (ii) the Eulerian approach [22]. The Lagrangian approach is individual-based and entails tracking a specific individual, while the Eulerian approach is place-based and deals with the probability of presence in a place and the change of this occurrence over time. Motivated by these two approaches in modeling movement behavior, we classify and give examples of the technologies, which can be used to collect data from mobile entities in the rest

[1] This chapter is partly based on:

[21] M. Baratchi, N. Meratnia, P. J. M. Havinga, A. K. Skidmore, and A. G. Toxopeus, "Sensing solutions for collecting spatio-temporal data for wildlife monitoring applications: a review," *Sensors,* vol. 13, pp. 6054-6088, 2013.

of this chapter. The general classification of these technologies is presented in Figure 2-1.

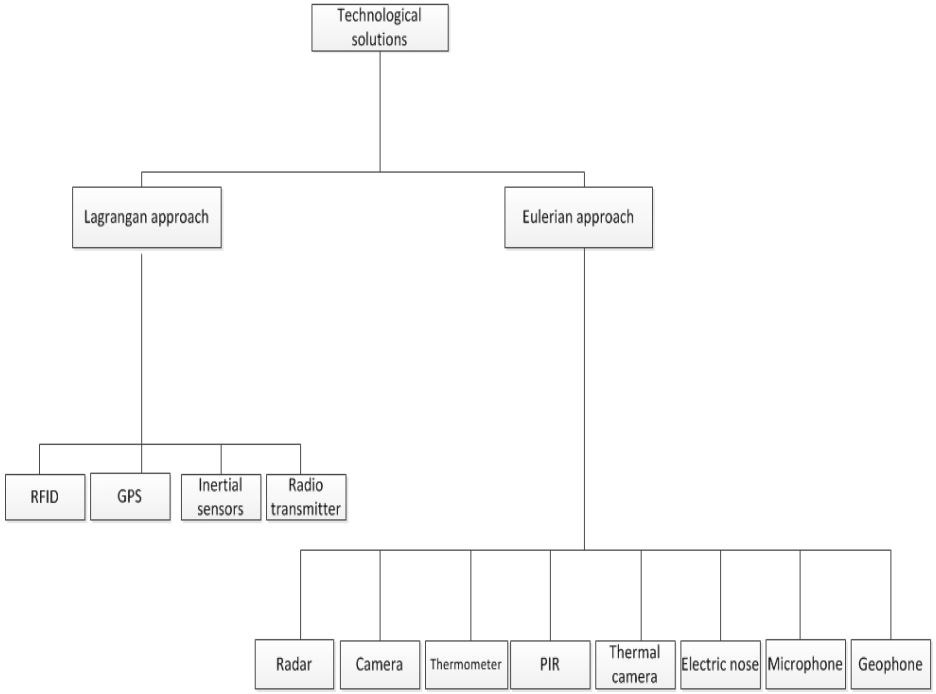


Figure 2-1 Taxonomy of the technologies used for collecting spatio-temporal data

2.2 Technologies for Eulerian approach

The essence of the Eulerian approach is modeling the pattern of space usage by an individual or a group [22]. A suitable data collection method for such studies should be able to record the data from a point in space, and interpret events that occur in that specific point. The technologies used for realizing such a modeling approach should collect the data inconspicuously and in a manner that it leaves the least effect on the mobile object. This implies that these technologies may have more impact on the environment than on the mobile object, as their long-term placement and difficulty of retrieving them after use may have implications for the environment. The sensing devices of this type are deployed in the environment and process the disturbances caused by the entity's presence in the environment. Although it is more difficult to extract spatio-temporal data using these solutions, when used to study animals, these technologies can provide more reliable results in terms of the impacts they leave especially on the animal's behavior, and their

health. They are also more reliable in human sensing as they do not require any effort from the subject in terms of carrying around a device and reduce the chance of missing data due to forgetting, or intentionally not carrying the device. The technologies, which are used to detect the above-mentioned disturbances, can be classified as passive and active. Active detection technologies such as radar and sonar can measure an entity's presence, range, velocity, or travel direction by how it modifies, reflects, or scatters an artificial sensing modality. Passive detection technologies simply record natural sensing modalities (visual, thermal, chemical, seismic, and acoustic) already present in the environment. In other words, active technologies both generate and receive a sensing modality while passive technologies only receive a modality. From the technical point of view, three factors are of concern when designing a system for Eulerian modeling. These factors are: (i) choice of modality, (ii) technical properties and (iii) data analysis techniques to extract the spatio-temporal properties. A challenging task in this case is data analysis since the measurements acquired by the sensing devices need to be further analyzed to extract information about the individual movement.

In the following sections, we review these modalities, type of sensors that can be used for each modality, and the type of data that can be acquired using them.

2.2.1 Radar (Echoes receptor/generator)

2.2.1.1 Echoes as a modality

There are a number of animals such as bats and dolphins, which use echoes for sensing their environment. Bats design their emitted waveforms according to whether they need to classify on the basis of micro Doppler effect (for dynamic entities such as insects or fish) or range profile information (for static entities such as flowers) [23].

Motivated by echolocation, active range RADAR (Radio Detection And Ranging), SONAR (Sound Navigation and Ranging), and LIDAR (Light Detection And Ranging) systems have been used for surveillance, entity recognition, and tracking. More recently, radar integrated with sensor networks has been found to be efficient when different categories of entities with individual identifiers exist.

2.2.1.2 Technology

Radar systems are operable in different frequency bands. The best applicable band of operation for low power systems is the ultra wide band (UWB). UWB radar systems use micro-power impulses rather than continuous narrowband transmissions. This makes them suitable for sensor data collection and tracking

applications. This type of radar is available in two broad categories, i.e., (i) pulse Doppler, and (ii) pulse echo. Pulse echo radar employs time of flight and is typically used as rangefinder [24]. Pulse Doppler radar operates based on the Doppler principle[3], and is primarily used for motion sensing (detecting location and velocity). There are currently commercially available radar sensors such as Bumble-Bee [25] compatible with wireless sensor boards.

While using radar as a sensor, the direction of motion of the entity can be critical, but a higher elevation angle can be used to avoid the difficulties experienced when the motion of the entity is perpendicular to the beam of radar [26]. Doppler radar has the advantage of being able to directly collect measurements of an entity's moving parameters. It can be used for creation of a fully automatic moving activity integral estimation procedure. Based on the frequency/wavelength ranges, or "bands", radar can penetrate barriers which obscure optical systems. For instance, UWB radar can be used in "see through the wall" applications. Other advantages of radar include operation in poor weather, day or night operation, and operation over long distances.

2.2.1.3 Data analysis

Most previous research in collecting spatio-temporal data by radar is based on micro-Doppler effect. The velocity of a mobile entity relative to an observer can be estimated by measuring the frequency shift of waves radiated or scattered by the object. This is known as the Doppler effect. In case of an articulated body such as a walking person, the torso and limbs each have their own velocity that changes over time. The modulation due to these movements are referred to as the micro-Doppler phenomenon [27]. The micro-Doppler effect provides signatures directly related to the kinematic information of the dynamic structural parts of a mobile entity and it offers new opportunities in classification of entities to different scales.

Various researches have been performed on extracting spatio-temporal properties from micro-Doppler effect. Examples include detection and classification of people when walking [28-30], finding the number of people present in the environment from their heartbeat patterns [31], distinguishing human from four legged animals [32], classification of different species by physiological characteristics [33]. Radar has been used to retrieve continuous spatio-temporal data on bird migration [34, 35], and to retrieve birds' flying characteristics such as height, velocity, direction,

[3] All mobile entities will exhibit a frequency shift from the transmitted signal to the received signal which is proportional to the speed of the entity in the direction of the radar

and density regardless of the time of fly. Echoes have also been applied to identify different fish species in depth [36].

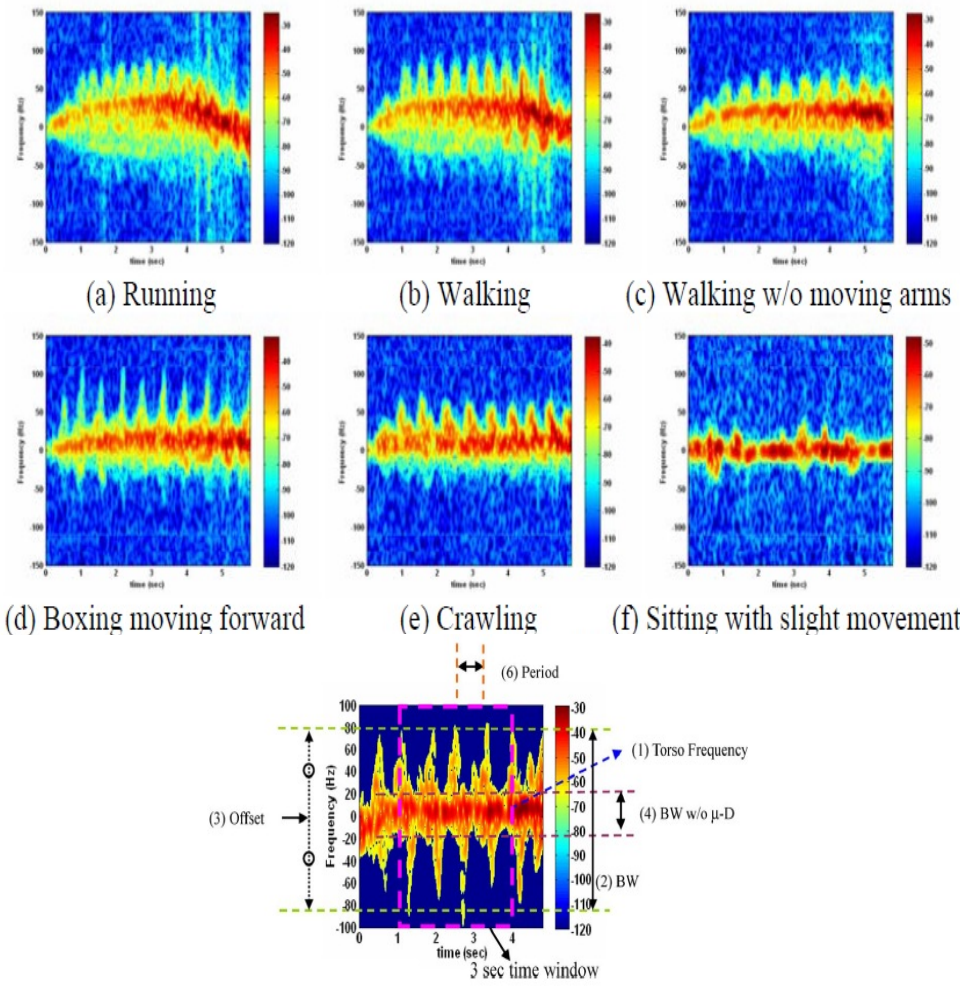
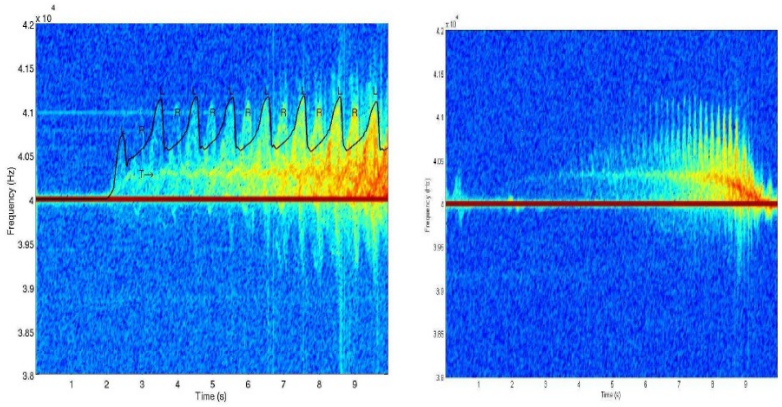


Figure 2-2 (1-f) Spectrogram of different human activities using micro Doppler signatures, (© 2009 IEEE. Reprinted, with permission, from IEEE Transactions onGeoscience and Remote Sensing, 47 (5), pp. 1328–1337) (g) Different features useful in activity classification (© 2009 IEEE. Reprinted, with permission, from IEEE Transactions on Geoscience and Remote Sensing, 47 (5), pp. 1328–1337) [37]

Other than identification and classification of animals at the species level, activities of the entity can also be classified based on micro Doppler signatures. Activity

recognition by radar has been studied in [37, 38] for human and [39] for lab animals. Figure 2-2 shows the time varying Doppler signatures of a person while performing different activities. Combination of a number of features, such as period of micro Doppler, bandwidth, frequency, and torso Doppler frequency (shown in figure 2-2) in each activity is different. For example, the period of the micro-Doppler in walking is longer than running. In the crawling motion, the torso Doppler is nearly zero and most of the micro-Dopplers are skewed toward the positive side with respect to the torso Doppler. Boxing while moving forward has a positive torso Doppler component in addition to the micro-Dopplers from the arms. Sitting has near zero torso frequency and small sporadic micro-Dopplers [37]. Figure 2-3 compares the spectrogram of a person and a dog. As can be seen, the period of the micro Doppler is clearly different between a human and a quadruped.



(a) (b)

Figure 2-3 (a) micro-Doppler gait signature of a person, (b) micro-Doppler gait signature of a dog walking towards an active sensing system (© 2007 IEEE. Reprinted, with permission, from IEEE proceedings of 41st Annual Conference on Information Sciences and Systems, pp. 627–630) [32]

2.2.2 Cameras (Visual receptors)

2.2.2.1 Visual modality

Visual interpretation of data can provide various types of information such as shape, size, and texture about objects. By extracting content information visually

from a scene captured by a camera, different parameters, such as quantity, species of the animal, and movement characteristic can be detected.

2.2.2.2 Technology

Integration of cameras and sensor networks has been studied in the context of multimedia and camera sensor networks. This has become possible with the availability of CMOS (Complementary metal–oxide–semiconductor) cameras. In comparison to traditional charged coupled device (CCD) sensors, CMOS image sensors are smaller, lighter, and consume less power. While regular CMOS sensors are still less energy efficient than the requirements of resource constraint applications, there is ongoing research to produce energy efficient hardware for use in camera sensor networks. Some examples of CMOS platforms are Cyclops [40], and CMUcam [41]. Hardware optimizations such as wake-up procedure, and dynamic voltage along with energy harvesting from the environment have made utilization of cameras in such networks possible. Each of the cameras within the nework can process data locally and then exchange relevant information with other cameras [42]. Furthermore, using a tiered network with different cameras (low/high resolution) in each tier, a more energy efficient system can be provided. In such cases, each camera can perform a distinct processing task with respect to its resources. One of the advantages of using normal cameras over technologies such as passive infrared sensors is their capability of identifying cold-blooded animals such as snakes. However, several factors such as foliage, lighting variations, and shadows can degrade the efficiency of camera-based systems.

2.2.2.3 Data analysis

Visual data can provide effective identification mechanisms. Individual entities can be differentiated from each other through their visual differences such as biometrics. Biometrics such as face for people and iris patterns, skin ridge prints, and nose prints for animals provide identity information. Coat patterns present in many species (such as cheetah, zebra, giraffe, orca) provide biometrics which are normally body sized and visible over distance. For example, automatic systems have been designed in [43] to identify zebras based on their coat patterns. It should be noticed that in such systems other parameters such as angle of view and the change in natural marks (due to aging, injury, pregnancy) can also introduce false identification. When necessary, paired cameras can be used to capture coat patterns from each side of the animal so that the complete coat pattern can be matched.

Other than natural markers, visual gait patterns can also be used to identify entities. Use of natural marks offers the advantage of detecting from long

distances. Visual silhouette based gait recognition is extensively studied for human identification. The gait recognition can be performed through model-based or model-free approaches. In the model-based approach, body and motion of the moving entity is modeled using a priori knowledge, while in the model-free approach gait appearance is considered without considering a-priori knowledge about the underlying structure [44]. To identify species type, model-based approaches are more applicable as they consider body shape, while more precise identification of animals at the individual level requires model-free approaches. In wildlife monitoring, subjects such as classifying species [45] and identification of individual cows [46] through gait recognition of visual images have also been studied. A common characteristic in all these methods is that they consist of two main stages, namely (i) a feature extraction stage and (ii) a recognition stage, in which standard pattern recognition techniques such as k-nearest neighborhood (KNN), support vector machine (SVM), and hidden Markov model (HMM) are used.

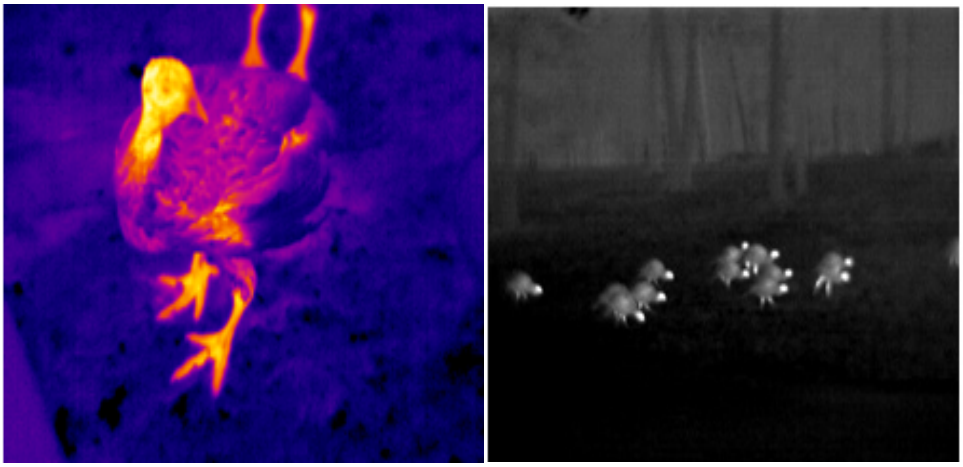
Vision-based localization methods can also be used to localize and track entities. Although entity movements follow a dynamically changing non-linear pattern, trajectories can be estimated by tracking low level features without depending on the success of the detection algorithm. Some previous studies have visually tracked flocks of birds [47], tracked animals using the gait patterns [48], and tracked by applying a specific interest model to the detected animal's face region [49]. Generally, entity tracking with a single camera involves the following steps: entity detection, classification, localization, and tracking frame by frame. Different features of the entities can be used for detection and tracking. Selection of the right feature for tracking is an important step. Some common features are color, optical flow, edges, and texture [50]. The vision-based localization, which is performed by a single camera, is limited to its field of view and the result will be in the image plane of the camera. In case more accurate result is needed, data from images of different cameras should be combined.

Images can also provide high amounts of information about entities' activities, behavior, and mutual interaction. Different methods, such as keeping the trajectory of the joints, action recognition with space-time volumes, or based on event and sub-events, can be used for the automatic single layer activity recognition [51]. Depending on the scheme used, the activity recognition can be performed on gesture, action, or interaction levels. Various vision-based activity recognition systems for wildlife have been designed. For instance, [52] have designed a visual system to determine five basic behaviors, i.e., sleeping, drinking, exploring, grooming, and eating of mice. In [53] a system has been proposed for detecting snake behaviors such as attacking. A complete survey on human activity recognition can be found in [51].

2.2.3 Thermal sensors (Thermal receptors)

2.2.3.1 Thermal modality

All objects emit infrared radiation at room temperature. Emissivity of a material is relative to its ability to emit energy by radiation. This electromagnetic radiation is a stream of photons, which are particles with no mass. Small changes in temperature may result in substantial amounts of emitted photons. Passive infrared radiation is very high in warm-blooded entities especially in mammals. Mammals have hot spots or specific thermal signatures, which can distinguish them from vegetation, constructions, and enable their detection. As Figure 2-4 suggests, factors such as the number of entities, is to some extent detectable visually using thermal sensors from the hotspots.



(a) (b)

Figure 2-4 (a) Infrared image of a Turkey, (b) Detectable hotspots of turkeys for analyzing crowd behavior (Both images are provided by The Snell Group), http://www.thesnellgroup.com [54].

2.2.3.2 Technology

Three types of sensors can be used to detect infrared radiations from animals. These sensors are (i) thermal imagers, (ii) passive infrared motion (PIR) detectors, and (iii) passive infrared thermometers.

Thermal imagers work on the basis of Infrared Thermography (IRT) which is the measurement of radiated electromagnetic energy. There are some limitations and factors that need to be considered when getting thermal images. Heavy feathers

and furs will reduce detectable radiation. The hair which covers the animal should be free of dirt and moisture, since the dirt on the animal changes the emissivity while moisture increases local heat loss [55]. To benefit from infrared radiation in low-powered networks, PIR motion detectors and passive infrared thermometers can be used. PIR motion detectors are devices which detect motion in their field of view by measuring changes in the infrared radiation from their surrounding objects. Due to their low cost and low operation power, these sensors are popularly used in wireless sensor networks for the surveillance purposes. Available wireless sensor node for ecological applications, are already equipped with passive infrared motion detectors as well as other environmental sensors (such as, the Mica weather board) [56]. Passive infrared motion detectors have been popularly used in combination with cameras in form of camera traps [57]. As shown in [58] strength of the output signal of the PIR sensor is not only determined by distance but also by speed of the mobile entity. Therefore, a PIR sensor network and simple signal processing algorithms can be used to obtain parameters needed for wildlife activity monitoring (in the covered area) such as direction, speed, distance, and counting the entities. A disadvantage of PIR sensors is that they can only detect presence in motion and presence of a static subject is not detected by them.

2.2.3.3 Data analysis

Since most warm-blooded entities have similar temperature ranges, the thermal signatures will not accurately discriminate between species. However they can show temperature-related states of the entities. As mentioned before, PIR sensors are widely used as presence detectors. Infrared cameras can provide information on specific species if the species has a discriminative shape (for instance, humans have been detected based on thermal shape [59]). Research has been performed in identification of individual human beings based on their thermal information (such as, face recognition by capturing facial physiological patterns using the bio-heat information contained in thermal images [60]). By considering and counting the hotspots in a thermal image the crowd behavior can also be analyzed [61]. Infrared thermography combined with infrared cameras has been used for many years to detect physiological states in entities. In [62], infrared thermography is used to measure the stress level caused by transportation in farm cattle. The basis of this research is that when the animal is stressed, changes will happen in heat production and heat loss in addition to blood flow response. Processing thermography images can provide information about the animal's stress level, health (such as, asymmetric heat distribution [63], abnormal surface temperature [64]), pregnancy, and any property related to the normal body temperature change.

2.2.4 Chemical sensors (Chemical receptors)

2.2.4.1 Chemical modality

All living creatures produce volatile compounds. Different environmental, genetic, and dietary circumstances, makes it improbable that any two organisms produce the same mixture of volatile organic compounds. On this basis, many animals can identify the members of their own group in a large group of other individuals [65].

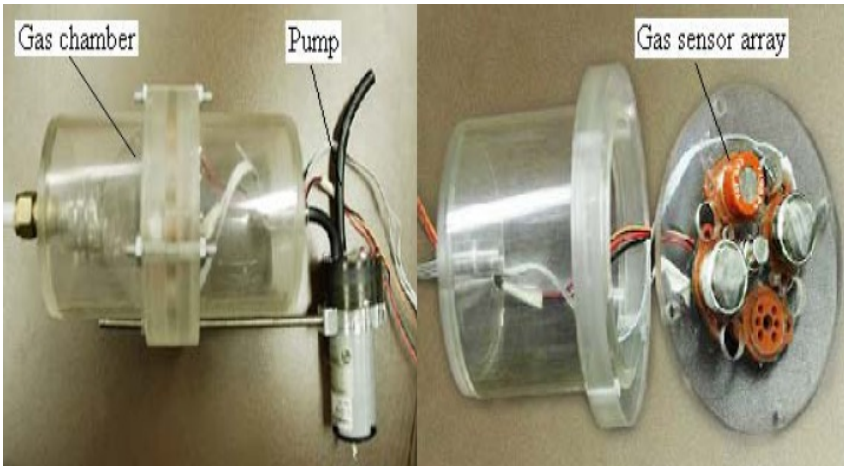


Figure 2-5 An electronic sensor node designed for the purpose of monitoring odorant gases and accurately estimating odor strength in and around livestock farms (Reproduced with permission from Simon X. Yang, Sensors, published by MDPI, 2009) [66]

2.2.4.2 Technology

Electronic noses have so far been used in various fields such as agricultural, biomedical, environmental, nutrition, medical, and military (A prototype is shown in Figure 2-5). Each electronic nose has two functions, *i.e.*, (i) sensing, and (ii) pattern recognition [67]. For sensing, chemical and gas sensors are used. By using a sensor array composed of different sensors, a wide group of simple and complex compounds can be identified. Different sensors have been used in the past to detect odors such as piezoelectric, conductivity, metal-oxide-silicon field effect transistor (MOSFET), optical fiber, and MEMS based sensors. Various types of artificial intelligence methods can be used for the purpose of pattern recognition to identify the smell [67].

2.2.4.3 Data analysis

Chemical cues are important phenomena in the biology of entities (especially, animals). Sex pheromones are well-known examples of chemical communication in different species. Producing electronic noses for detecting pheromones has been explored for insects. Biosensors have been produced by real moth antennas to extract electrical signals which are produced in existence of other moths [68]. Pheromone detection of insects along with their specification such as gender and species from their pheromones is studied in [69, 70]. Commercially available electronic noses (such as, Cyranose [71]) are used to detect stink bugs and stink bug induced damage in cotton product under laboratory and field conditions.

One of the important issues in designing an electronic nose is that the combination of volatile compounds of the odors should be detected beforehand. This procedure is normally done through gas chromatography and mass spectrometry [72]. It should be noticed that for typical sensors, the detection threshold of odors are extremely low. Therefore, detection of a very low amount of gas compound in a large environment is challenging. To be used around livestock farms where the volatile organic compounds are well known, commercially available gas sensors and high intensity of gas make the compounds easily detectable [73]. However, although detection, tracking, and identification of specific entities (especially mammals) is possible, extensive research should still be performed on pre-analysis of the compounds.

Electronic noses have been used to identify human beings from their body odor [74]. Although individuality in body odors has been described in a variety of mammals [75], the electronic noses which have been designed so far for animal studies mainly focus on purposes other than identification, such as controlling gas level around farms. Moreover, the changes in the individual compounds due to effects of physiological and seasonal changes should be considered for designing a system for classification [76].

Since the composition of the compounds in entities' odors change with various physiological parameters (such as, their health condition, age, and estrous cycle), the electronic noses can be used to detect events that cause changes in the compounds. In such cases, placement of electronic noses in places with rich sources of known volatile organic compounds is more logical. For instance, in [77] e-noses show a correlation between evolution of the odor with animal activities during the day and with their age around the farm.

2.2.5 Microphones (Acoustic receptors)

2.2.5.1 Acoustic modality

Humans use sound for communication and animals produce sound for different purposes such as defending territories, attracting mates, deterring predators, navigation, finding food, and maintaining contact with members of their social group. Sound production by entities can be divided into two categories, i.e., (i) non-incidental sounds which are used for communication purposes, and (ii) the incidental ones which are the result of their activities [78]. These sounds can be used to detect the presence or to identify a species. An important challenge in acoustic sensing is that ambient noise and anthropogenic sounds can make the acoustic signal processing difficult.

2.2.5.2 Technology

Microphones are acoustic transducers, which produce a voltage proportional to the received acoustic pressure. Microphones work at different frequency ranges and the right frequency of operation is chosen by considering the type of sound produced by animals. Generally, animal voices are categorized as sonic, infrasonic (elephants), and ultrasonic (bats and dolphins). This fact determines the type of microphone needed to capture the sound of interest. Directional microphones can be used for capturing sound from specific directions [78]. Due to extensive amount of research done in the field of acoustic sensor networks, microphones are already part of commercial wireless sensor network boards (such as, Mica sensor board [79]). Furthermore, specifically designed platforms for acoustic sensing networks are available. For instance, the Acoustic ENSBox system provides a platform for developing deployable prototypes of distributed acoustic source localization and sensing applications [80].

2.2.5.3 Data analysis

Different algorithms allow us to distinguish entities in terms of species, gender, age groups, and individuals by automated signal detection and classification based on features extracted from signal frequency, mel-frequency cepstral coefficients, or signal energy distribution [78]. Both voiceprints and behavioral sounds can be used for the purpose of classification of entities in different levels. Human voice recognitions is a well studied topic [81]. Regarding animal voice recognition, most of the researches focus on species identification [82-84] and specifically on classification of bird species based on bird song [85, 86]. Individually distinctive acoustic features have been demonstrated for a large number of birds, mammals,

cetacean, and amphibians. Various studies have been performed to identify individuals based on these voiceprints (whooping cranes [87], African wild dogs [88], eagle owls [89] and ant-thrushes [90]).

Other than non-incidental voiceprints, incidental sounds may entail information about the identity of entities. To the best of our knowledge, there is no work on classification of animals based on their acoustic gait signature. The footstep sound analysis which has been used for human identification [91] may also be applicable for animals. Studies show that footstep convey information about personality, age, and gender [92].

An entity's position can be determined using an acoustic localization algorithm. Sound signals are omni-directional and have a uniform attenuation model. Microphone arrays can be used to provide efficient localization of the animal without having a line of sight to the animal. Through localizing entities acoustically, information about their interaction, count, and population distribution patterns can be extracted [78]. For implementing such systems several sensor array nodes need to be located at points surrounding the entities and their territory (animals) or stay-point (humans) and when detectable sound is made, the most likely location will be estimated by the sensor array nodes. Acoustic source localization methods make use of three types of physical measurements, i.e., time delay of arrival (TDOA), direction of arrival (DOA), and received signal strength or energy [93].

2.2.6 Seismic sensors (Seismic receptors)

2.2.6.1 Seismic modality

Producing seismic vibrations on a substrate is a means of communication in different species (invertebrate and vertebrate). An interesting seismic effect produced by humans and legged animals, is footstep. Animals have distinctive gait patterns (4-beat gait, 2-beat gait, and canter; 3-beat gait and some unnatural walking patterns, collective walk or working walk [94]). Footsteps produce seismic effects that pass through the ground. These effects propagate away from the source as seismic waves. These waves are classified into two categories, *i.e.*, (i) body waves (33%), which travel towards the interior of the earth, and (ii) surface (Rayleigh) waves (67%), which travel near the surface [95]. Most of the researches focus on surface waves to provide classification systems of entities.

A number of factors determine the performance of vibration detection. These factors include the resonance frequency of a vibration, the frequency of impacts

(footsteps), the strength of the wave (enforced by entities weight), the friction of the medium, the underlying geology [96] and noise sources such as wind and cultural noise (undesirable noise produced by human urban activities). Wind noise may be coupled into seismic ground sensors by either direct or indirect (adding seismic noise through shaking trees) means [96].

2.2.6.2 Technology

Surface waves are measured by two types of transducers; (i) geophones, and (ii) accelerometers. Geophone is a device that changes the velocity of ground vibrations into a voltage. Geophones are normally buried to be safe from animal's destruction. As stated in [96], these devices show both low-frequency (10 Hz, 14 Hz, 28 Hz, and 40 Hz) fundamental resonance and high-frequency spurious resonance (25 times of the fundamental). For detection of vibration generally only frequencies that lie between the fundamental and spurious resonances should be used [96]. Most energy of human footsteps is between 10 and 100 Hz repeating with a frequency between 0.9-3.5 Hz [97]. As the authors of [98] have characterized, the main part of the footstep signal energy for a distance more than 6 meters, is usually bellow 100 Hz. These researchers have shown that, as the distance between a walking person and a seismic sensor increases from 6 to 60 meters, the signal frequency maximum moves closer to 10–16 Hz [98]. The resonant frequency range of the entities footfall and these facts should be employed in choosing the right geophones sensor. The second type of seismic sensors is accelerometer, *i.e.*, a device that changes the acceleration of the ground vibrations into a voltage. Accelerometers only show high-frequency resonant frequency (over 1,000 Hz [96]) and this makes them unpopular in footstep detection [96]. However, when the acoustic waves are transferred through the substrate, these vibrations can be detected with an accelerometer. For instance, high-frequency acoustic waves can also be detected with accelerometers while passing the substrate [99]. Accelerometers have also been part of the Lagrangian modeling systems to collect movement data from mobile entities [100].

2.2.6.3 Data analysis

Numerous seismic surveillance studies have been performed to classify entities such as, vehicles and soldiers, compute their bearing, and their velocity based on seismic features [101, 102]. Humans and animals can also be detected based on their footstep-generated seismic waves. The signature of footstep is in form of sharp "spikes" and distinguishable from other noises [103]. By measuring the seismic signals using a seismic velocity transducer presence of the moving entity

can be detected. Different features and statistic characteristics of signal can be used for detection and distinguishing the animal. Afterwards, the signals can be classified using artificial intelligence methods. For example, spectral analysis for discriminating between seismic events caused by entity's footsteps, cadence [94] (the interval between events (footsteps)) and kurtosis [104] (degree of peaks in a distribution) are used for footstep detection. In the domain of human sensing, seismic waves have been used to detect presence of humans [105], the number of people, and direction of travel [106], tracking and bearing estimation [107]. Mainly the seismic studies that consider animals have focused on differentiating between two categories (bipedal (human) and quadruped (animals) [94, 108]). Few wildlife studies have focused on problems such as detection or classification of animals. For instance, [109] has investigated a number of problems, such as detection of elephants from a distance of 100 meters, counting the number of individuals, and differentiating their species from other species. The species (to some extent) might also be detected based on the influence field (the number of sensors that sense the vibration [110]). Seismic communication, foot-drumming, distinctiveness of footsteps in terms of gender of animal have also been studied [111]. Moreover, in low-noise environments underground organisms can be detected. For instance, in [99] the acoustic waves produced by a colony of ants underground is detected and classified by geophone from a distance of few centimeters.

2.3 Technologies for the Lagrangian approach

As stated before, Lagrangian based technologies are in form of a device carried by the mobile entity. Generally, when choosing a Lagrangian based technology some general requirements need to be kept in mind:

- When used for studying animals, the device should preferably weigh less than 3-5% of the animal's total bodyweight (no more than 10% for terrestrial mammals [48]).
- The device should have a relatively long lifetime so that it is not needed to be collected again before the necessary amount of data is collected.

Compared to the technologies for Eulerian approach, the issues that should be greatly concerned in the Lagrangian approach are the choice of the hardware and retrieval of data from the entity. When used on wild animals, it is important to have a mechanism for automatic retrieval of data from the device through single/multi-hop networks, as the change of recapturing the entity is small.

In what follows, we review a number of technological solutions used in the Lagrangian approach for collecting spatio-temporal data and their integration with wireless sensor networks.

2.3.1.1 Radio Frequency Identification

2.3.1.2 Technology

Radio Frequency Identification (RFID) is a technology designed for storing and retrieving data by using electromagnetic transmission. Nowadays RFID is being used as a means of enhancing data handling tasks [112]. The RFID systems consist of two main components, *i.e.*, tags and readers. Each tag has a memory that stores an Identification number. This memory can also store additional data such as environmental parameters (temperature, and humidity). The reader (including an antenna) reads and/or writes data to tags through electromagnetic transmissions. RFID tags have been used to study various entities (birds [113], reptiles [114, 115], amphibians [116], mammals [117, 118] , and humans [115]).

RFID technology is originally designed for retrieving identity but it can be used to retrieve location as well. After detecting and identifying the moving tag, different types of algorithms can be used to calculate the current location of the tag, relative to the readers' location. Localizing techniques for RFID tags are known as RF based localization which lie on the same principles of the ones for wireless networks [119]. To save power in an event-based manner, the procedure of tracking the object can be done in a predictive manner to activate the readers, which are in idle state.

Tags: RFID tags themselves may be active, passive, or semi-passive. They may be supplied in a variety of forms and work based on ISO standards [144].

- **Active tags:** Active RFID tags are equipped with their own independent power source. Thus, they are able to transmit a stronger signal which can be accessed by readers placed in far distances. These tags operate at higher frequencies, commonly 455 MHz, 2.45 GHz, or 5.8 GHz. Based on the frequency of operation, readers can communicate with active RFID tags from a distance of 20 to 100 meters [120]. The onboard power source makes the active tags larger and more expensive. To increase the lifetime of tags, they can be switched to sleeping mode until they come in range of a receiver.
- **Passive tags (Passive integrated transponder, PIT tags):** For monitoring purposes passive tags are made available in different forms (implants, or ear tags).). They consist of three parts, i.e., (i) an antenna, (ii) a chip attached to the antenna, and (iii) encapsulation. Passive tags do not have an internal power source and the reader is responsible for powering them. When these tags come within the reader's range, they receive an electromagnetic signal from the reader, and the energy is stored in an on-board capacitor. Because of their small size and weight, they are useful to study small entity movements with less disruption. Furthermore, without having a power supply they will last for

the life of the entity. This technology is very popular for tagging fish [121] and it has even been used for studying ants [122]. Passive tags typically operate at 128 KHz, 13.6 MHz, 915 MHz, or 2.45 GHz frequencies with the read ranges between a few centimeters to 10 meters [123]. Factors such as frequency of operation, antenna dimensions, and modulation type determine the read range. Since the water in living tissues absorbs high frequency photons, most of the passive implants designed for identifying entities operate in low frequency (125-kHz and 134.2-kHz), while passive external tags work in higher frequency ranges. Passive tags are small and cheap themselves but the readers are relatively big and noticeable and for having better detection range, the size of the antenna increases extensively. Figure 2-6 shows a number of passive RFID tags used for monitoring live organisms.

(a) (b)

Figure 2-6 (a) A radio tagged ant (Reproduced with permission from RFID Journal, 1 August 2009. Image, and copyright held, by, Nigel R. Franks)[122], (b) Passive RFID implants (Image provided by Biomark Inc.)[124]

2.3.1.3 Integrating RFID technology with wireless sensor networks

RFID technologies work on the basis of single hop communication between a reader and a tag. Integration of RFID readers within sensor networks has improved functionalities of these systems both by enjoying the capabilities of wireless communication and retrieving additional data. In this type of integration, the RFID

readers are provided with multi-hop communication capability and other type of sensors. They will be able to communicate with each other wirelessly and sense environmental parameters and disturbances. This type of integration is an ideal solution for collecting various types of data through one network. Readers provide identity information. Other types of data, such as environmental and activity related data, are collected by the sensors (For instance, a camera) on the RFID reader. These data are then transferred in a multi-hop fashion. The SkyeRead Mini M1 made by SkyeTek is an example of an RFID reader (for reading 13.56 MHz RFID tags) designed to mate directly with the Crossbow Mica2Dot sensor mote [125]. This kind of RFID reader can be used with various types of passive ear tags. In [117] a hybrid detection node is designed by integrating RFID readers with Tmote Sky motes for collecting spatio-temporal data from badgers. Authors of [118] have interfaced the tag readers with Fleck wireless sensor network nodes to track the movement of farm animals near the readers.

2.3.2 Global Positioning System (GPS) Technology

2.3.2.1 Technology

Global Positioning System (GPS) is a widely used localization and tracking system in various domains and especially in wildlife monitoring [126, 127]. Currently, almost any type of smart phone is equipped with GPS sensor. This way, context-rich datasets from humans are collected. The GPS system components are a space segment (24 satellites), a control segment (network of ground based stations) and a user segment (receivers that convert satellite signals into location estimates). The receiver acquires signals from at least three satellites to obtain 2D positions (4 satellites for getting 3D positions). There are two approaches to retrieve the data collected by the user segment, i.e., offline and online. In the offline approach, the "Store on Board" user segment should log the data, which will be later either retrieved after the user segment is collected or manually downloaded through handheld receivers. To access the data in real-time, different telemetry systems such as the ARGOS satellite system, radio telemetry transmitters, and GPRS have been used so far. Recently wireless sensor networks have been used to transfer GPS data as well. It should be mentioned that due to energy consumption and size restrictions, currently, it is inevitable to have the data collected offline in certain cases (For instance with GPS collars shown in Figure 2-7) otherwise the system will not be able to meet the average life-time requirement. Today several commercial GPS equipment such as, Televilt [128], Northstar [129], Lotek [130], E-obs [131], Microwave [132] are available, which provide more than just positional data. These devices are designed for almost any entity (birds, mammals, reptiles, and rodents) and depending on their size and weight, these GPS receivers can be equipped with

bidirectional transmitters (work sometimes up to 400 meters), a set of sensors, contact loggers (to log the contact between two animals) and a power source. Also, having considered wireless sensor networks requirements, Fleck family boards [133] have been designed for the wildlife monitoring purposes. When used for wildlife monitoring, other than the risks that it may have on the animal's health and normal activities, another limitation of the GPS devices is that in certain conditions the GPS receiver will not be able to receive enough satellite transmissions. These conditions include (i) atmospheric conditions like cloud cover, humidity, (ii) biophysical conditions like under dense foliage, steep terrain or buildings, (iii) indoor applications, and (iv) changes in the orientation of the antenna due to animal behavior).

2.3.2.2 Integrating GPS technology and wireless sensor networks

When using the GPS as sensors of wireless sensor networks, generally two types of communication architecture will be possible:

- **Mobile node to mobile node communication:** In this type of communication, mobile wireless sensor nodes carried by mobile entities are equipped with GPS modules. These mobile nodes are the only constructors of the system and the data they collect should be passed from a mobile node to another mobile node until it reaches the gateway. In this case, the advantage of wireless sensor networks over the previous telemetry mechanisms is providing the capability of extending the transmission range and reducing the power consumption of the GPS device through opportunistic routing protocols. In this form of architecture, if none of the entities (GPS receivers) comes to the proximity of the gateway, no data is transferred. So far, various studies have transferred GPS data with the help of wireless sensor networks using this architecture [134-137].
- **Mobile node to static node:** In the second type of communication, other than the mobile nodes, equipped with GPS modules, there is a ground based wireless sensor network that collects data from mobile nodes. This form of communication alleviates the problem of sporadic connectivity of the previous one. In [138], authors have designed a system in which GPS devices fitted on animals are able to communicate with an array of static nodes to return data to a central base station.

(a) (b)

Figure 2-7 GPS collars: (a), GPS collar designed for small mammals, (b) GPS collar designed from medium-large sized mammals (Images provided by Lotek Wireless Inc. [130]

In both of these type of communications, the system can be devised to be delay-tolerant which means that data can be buffered in a node before it comes to the proximity of another node. However, they lack of applicability when animals travel far distances and do not return to the connected area.

2.3.3 Inertial sensors

2.3.3.1 Technology

By using inertial measurements such as speed and direction, track of the mobile entity can be estimated. Inertial measurement unit sensors such as accelerometers, gyroscopes, and magnetometers can be used for this purpose. Accelerometers, gyroscope, and magnetometer measure acceleration, angular velocity and magnetic field respectively. Metal objects, external magnetic field and gravity are sources of error in inertial sensors.

By knowing the mobile entity's initial location, its next position can be estimated having the velocity or acceleration during that time interval. This scheme is referred to as Dead reckoning. Dead reckoning is a complementary addition to GPS systems for preserving energy and retrieval of localization data when a GPS device is not responsive (e.g. due to cloud cover). However, this method is subject to accumulative error. Dead reckoning has been used frequently for indoor tracking of human and pedestrian navigation [139, 140].

2.3.3.2 Integration with wireless sensor networks

If inertial sensors are embedded on sensor network nodes attached to mobile entities' (especially feet), they can be used for the purpose of localization. For instance, authors of [141, 142] have designed a system for localization of rats in underground burrows to estimate their steps with inertial sensors. On the exit of the burrow the data (the number of steps and their direction) is transferred to a static node, through which the path and structure of the underground borrow is constructed. This form of tracking, although not very accurate, can be effective when a network of static nodes is used to update the position of the mobile sensor nodes. Inertial sensors have also been popularly used in activity recognition and behavior analysis systems for humans [143] and animals [100, 144, 145].

2.3.4 Radio transmitters

2.3.4.1 Technology

Using radio transmitters for tracking has long been popular. Radio telemetry systems consisted of a device carried by a mobile entity, which transmitted a radio signal. The position of the mobile entity could be retrieved by triangulation. Later some additional sensors were added to those devices and additional parameters could also be sent through radio telemetry. The tracking range in radio telemetry methods is limited by radio range of the UHF (ultra high frequency) or VHF (very high frequency) and the devices used are relatively power hungry.

2.3.4.2 Integration with wireless sensor networks

Recently more efficient systems have been designed using wireless sensor networks. The wireless motes consisting of a transceiver, memory, and a micro-processor (no other sensors) are carried or fitted on mobile entities. By doing so, the position of the entity is estimated through radio communication between these mobile nodes and a static gateway, without the help of any additional sensors. For instance, in [146] entities register their presence when they come into the transmission range of a gateway and the received signal strength is used at the gateway to estimate the location of the entity. This form of localization and identity retrieval is in essence a better alternative than using active RFID tags, which register the presence of the tagged entity near the reader. In this case there is no need to interface a commercially designed reader with the gateway (i.e., a wireless sensor network node).

2.4 Discussion

In this section, we compare the aforementioned technologies in terms of data they provide and their applicability for studying mobile entities' behavior. Tables 2-1, 2-2, 2-3 provide a comparison of the technological solutions mentioned previously from different perspective. We compare the aforementioned sensing technologies in terms of their spatio-temporal data collection capability and do not consider other data types such as behavioral and physiological data. Adopting the definition of spatio-temporal properties in the domain of human sensing [147], we identify and define track, identity, species type, location, presence, and time as spatio-temporal properties needed for studying mobile entities (humans and wildlife).

- **Track:** is the most comprehensive spatio-temporal feature, which can be simplified as the location of an individual or a group over time. Therefore, the three properties of location, time, and individual/group identity are needed to maintain track of an individual or a group.
- **Identity:** is a global unique identifier assigned to an entity. It may be the permanent ID of a tag, or a detectable biometric or sign, which can show the individual identity of a moving entity. When assigning a unique ID to an entity is not possible, a temporary identifier may be used.
- **Species type:** The species type can be considered as a low level identity that can be assigned to mobile entities. We can also consider it as a sub-property of identity.
- **Location:** Localization is determining the location of the mobile entity. Whenever coarse-grained coordinates are acceptable, localization can be reduced to presence detection. Otherwise, a separate mechanism should be used to localize to a finer grained scale. For example, a single microphone can be used for detecting the presence of an organism through its voice but it is not enough to detect its location, the position which can be inferred in this way is only an estimate around where the microphone is placed which might be a relatively large area. For having more exact location coordinates a microphone array is needed and the fine coordinate can be calculated through various acoustic source localization schemes.
- **Presence:** Detecting presence is the ability to detect the presence of an organism in a field. Detection alone without classification of the species, detecting the identity, or the number of subjects may only provide limited amount of information. However, technologies that can only detect presence of a mobile entity can be used as an input to other more powerful but yet more resource consuming technologies.
- **Time:** is an essential property among spatio-temporal properties. There should be the possibility of assigning timestamps to properties, which are detected to make them meaningful for later analysis. The frequency of

timestamps is relative to how frequent other properties should be sensed so that the integrity of data is not lost.

2.4.1 Comparison of technologies based on information they can provide

In Table 2-1 we compare the previously reviewed sensors in terms of their ability to provide the aforementioned spatio-temporal features as well as their invasiveness and active/passive mode of operation.

	Technology	Presence	Species type	Identity	Location	Invasive	Type *
Eulerian	Radar	+	+	+	+	-	A
	Geophones	+	+	+	+	-	P
	Microphones	+	+	+	+	-	P
	Thermal cameras	+	+	-	+	-	P
	PIR	+**	-	-	-	-	P
	Thermometers	+	+	-	-	-	P
	Electronic noses	-	-	-	-	-	P
	Cameras	+	+	+	+	-	P
Lagrangian	Passive RFID tags	+	+	+	+	+	P
	Active RFID tags	+	+	+	+	+	A
	GPS	+	+	+	+	+	A
	Inertial sensors	+	+	+	+	+	P
	Radio transmitters	+	+	+	+	+	A

Table 2-1 Comparison of sensing technologies in terms of spatio-temporal properties they provide. * In the column Type, P represents (Passive), and A representes (Active). ** (Only in motion)

The field under a property is marked with "+" if the results of previous researches show that this technology is applicable for the purpose of obtaining spatio-temporal properties in mobile entity monitoring applications. Otherwise, it is marked with "-". Most of the identified technologies for the Eulerian approach can detect presence. If the purpose is to detect the presence of a warm-blooded entity, PIR sensors are particularly useful. Radar, geophones, and microphones are capable to extract all above-mentioned spatio-temporal properties. By performing image processing techniques on the images collected by thermal cameras some spatio-

temporal properties can be extracted. Since most of the visual biometrics are not shown in thermal images, the applicability of thermal cameras in identification is low. As mentioned above, electric noses are theoretically able to identify individual/species but the plausibility of current electric noses is not yet proved for monitoring mobile entities (due to the low amount of organic compounds).

The Lagrangian based technologies are more useful in extracting spatio-temporal properties from individuals. RFID tags work on the basis of transmitting an ID number. The devices equipped with GPS, radio transmitters, and inertial sensors can be programmed to send a unique ID number periodically. Being able to extract the individual identity, presence and species type of the tagged individual can also be inferred. Therefore, all these technologies are marked with + indicating their ability to provide presence, species type, identity, and location information. Although RFID tags and radio transmitters are not designed to measure location, location information can be calculated from specifications of the signal transmitted between the device on the entity and readers (or receiver). A single device equipped with GPS and inertial sensors estimates location and there is no need to calculate the location by taking the measurement of other devices into account.

The major difference between technologies used by the Lagrangian and Eulerian approaches is the applicability of the Eulerian based technologies for extracting spatio-temporal properties from **any** entity in their field of view, while technologies used by the Lagrangian approaches are only useful for extracting these properties from the **tagged** entities.

Tracking is the only spatio-temporal property, which is not mentioned in Table 2-1, because it has to be inferred from combination of other properties. For instance, a network, which can continuously detect presence of a moving entity, or a sparse network with identification capability, can both provide tracks. Choosing which spatio-temporal property to use for tracking depends on the data needed for the purpose of later analysis. For instance, for monitoring migration of specific species, species recognition with coarse-grained localization is enough to provide the necessary tracks since individuals stay in the flock. Therefore, sometimes the group track might be more important than the individual track.

2.4.2 Comparison of technologies based on different performance metrics

It is difficult to compare all of the aforementioned technologies based on metrics such as operational range and power consumption. The reason is that some of these sensors are not commercially established and data regarding these properties are not available for them. However, by reviewing previous research, a number of general conclusions can be drawn.

Regarding the power consumption, passive and active mode of operation impacts the energy efficiency. Active mode of operation requires generation of a sensing modality and is, therefore, more energy consuming. Some passive technologies such as visual and thermal cameras are also not energy efficient since the visual data type is considerably large, and require more memory and processing power. Among the sensors used in the Eulerian approach, PIR is known to be the most energy efficient with lower cost. It is widely used in low power surveillance systems where the area is covered with a large number of nodes with short coverage range. However, the information extracted from these sensors is limited. Camera, microphones, and radar are more suitable for longer-range operations depending on the characteristic of the entity. These sensors also provide more information from the studied entity. It is also important to consider the line of sight requirement of the sensor. Among these technologies, thermal and ordinary cameras, PIR and thermometers need a direct line of sight to the entity. Based on the frequency of operation radar can be used when physical barriers exist between the entity and the sensor.

In Lagrangian based approaches, use of GPS is specifically recommended for long-range localization purposes due to its global coverage and accuracy (relative to the covered area). However, GPS is considerably power hungry. In order to meet long lifetime requirements, it is sensible to use duty cycling and low power inertial sensors to measure the approximate location when the GPS module is off. The shortcoming of the technology relates to the fact that its applicability has been only proven for heavy moving entities, as a GSP-enabled device with a lifetime of few days/weeks is relatively heavy. Active RFID tags and short-range radio transmitters are more useful when the weight requirements are critical and for animals with limited range of spatial activity. Although passive RFID tags are small and light, their extremely low detection range limits usage in an automated system. Furthermore, when using RFID tags energy provision for an automated reader remains a challenge.

In Table 2-2, we compare the technologies in terms of the (i) outdoor disruptions that degrade the performance of a sensor, (ii) amount of processing which is required for extraction of each of the spatio-temporal properties mentioned previously in Section 2-4 (P (presence), S (species type), I (Identity), L (location)), (iii) commercial establishment of platforms for being used in wildlife monitoring, and (iv) invasiveness with respect to their effect on the entity under study. Under the column representing the processing required for each of the previously mentioned spatio-temporal properties, we mark a technology with L (Low) when low amount of computation is needed on the device and H (High) is used otherwise. It can be seen that although the technologies used for the Eulerian approach can provide more

information, extraction of these information requires considerable amount of offline data analysis.

Approach	Technology	Outdoor disruptions	Processing required				Commercially Established	Invasive
			P	S	I	L		
Eulerian	Radar	Smoke, dust, humidity	L	H	H	H	-	-
	Geophones	Cultural noise	H	H	H	H	-	-
	Microphones	Wind, background acoustic noise	L	H	H	H	✓	-
	Thermal cameras	Smoke, dust, humidity	H	H	H	H	✓	-
	PIR		L	-	-	-	✓	-
	Thermometers		H	H	-	-	✓	-
	Electronic noses	Humidity, air quality, wind	-	-	-	-	-	-
	Cameras	Unsuitable lighting conditions	H	H	H	H	✓	-
Lagrangian	Passive RFID tags	-	L	L	L	L	✓	✓
	Active RFID tags	-	L	L	L	L	✓	✓
	GPS	Cloud cover, heavy foliage, indoors	L	L	L	L	✓	✓
	Inertial sensors	Metal objects, external magnetic field and gravity	L	L	L	L	✓	✓
	Radio transmitters	-	L	L	L	L	✓	✓

Table 2-2 Comparison of sensing technologies based on a number of performance metrics. P (presence), S (species type), I (Identity), L (location)

2.4.3 Comparison of technologies based on the subject of study

Table 2-3 summarizes the aforementioned technologies based on their usage in research studies. It should be mentioned that, references given under each entity type are only those included in this chapter. It is also worth mentioning that not all of these technologies have been used in combination with wireless sensor networks. However, they have the potential to be used in such networks. Marking "-" under species type means that the corresponding technology is not popularly used for studying that species.

Technology	Animals					Humans
	Mammals	Birds	Amphibians	Reptiles	Fish	
Radar, ultrasound	Dog and horse [32]	Migratory birds [33-35]	-	-	[36]	[26, 28, 30, 32]
Camera	Lion [49], Zebra [43] Cows [46] Quadrupeds [45] [49] Rats [52, 148]	Migratory birds [47]	-	Snakes [53]	-	[50, 149-152]
Infrared technologies	Cows [62, 64] Quoll[57] Zoo mammals [63]	Ostrich [63]	-	Lizzard [153]	-	[59, 62]
e-nose	Livestock [73]	-	-	-	-	[74]
Geophone	Quadrpeds [94] Elephents and large mammals [109] Mole rat [111]	-	-	-	-	[101-103, 105, 106]
Microphone	Lycaon pictus [88]	Bald Eagles [154] Crane [87] Ealge owl [89]Ant-thrushes [90]	Cane-toad [84, 155] Anurans [82]	-	-	[91, 92]
RFID	Badgers [117] Farm animals [118]	Tern [113, 156]	salamanders [116]	Corn snake [114]	**[121]**	[115]
GPS	Livestock [134-138]Zebra [134] Caprocorn [13] Mountain lions[135]	Migratory birds [126, 127, 157]	-	-	-	[158]
Inertial sensors	Rats [141, 142] Cows [104, 162] Different species[161],	-	-	-	-	[139, 140].
Radio transmitters	Cows [146]	-	-	-	-	-

Table 2-3 Summary of the technological solutions with respect to the studied animal

2.5 Summary

Wireless sensor networks provide additional advantages over previous telemetry methods in collecting spatio-temporal data, which make them suitable for various mobile entity sensing applications. Two types of movement modeling are possible using these networks (i.e. Lagrangian, and Eulerian). Collecting spatio-temporal data with wireless sensor networks especially for the Eulerian approach has various unexplored domains, as the number of research in this domain is limited and relatively sparse. To provide suitable outcome for the movement modeling, high level improvements are still needed both in terms of software (for extracting spatio-temporal properties) and hardware (for sensing). Different well-tested schemes in the domain of human sensing are not yet applied in wildlife monitoring approaches. For instance, gait biometric, a well-explored biometric in human-sensing, is not considered seriously in wildlife monitoring projects. Gait pattern has the potential to be detected with different sensors (radar, seismic, visual, and acoustic) for extracting species type, identity or maybe physiological state information that it conveys. Furthermore, in some domains such as chemical sensing, the technology still has to improve to be able to provide necessary spatio-temporal data. For instance, although there is evidence in favor of measuring different physiological states and identity level distinction through chemical disturbances, the technology that is usable in conjunction with wireless sensor networks is still far away from use.

With respect to the previously mentioned advantages for Lagrangian technologies, which provide high quality spatio-temporal data, wide-range outdoor coverage, less need for infrastructures, and accuracy, in the rest of this thesis we use data acquired using these technologies. More specifically, we base our data analysis solutions on spatio-temporal GPS datasets. However, the techniques we propose are not specific to the Lagrangian based technologies, many of the algorithms we propose are also applicable to spatio-temporal data acquired from other technologies.

Chapter 3

Mining periodic behavior from streaming mobility data[4]

Both humans and animals pursue periodic activities in their lives. Extraction of periodic behavioral patterns hidden in large volume of mobility data helps in understanding the dynamics of activities, interactions, and life style of mobile entities. The ever-increasing growth in the volume and dimensionality of mobility data on the one hand, and the resource constraints of the sensing devices on the other, have not only made accurate pattern recognition a challenge, but it has also made low complexity, and low resource consumption important requirements for periodic pattern recognition algorithms. In this chapter, we propose a method for extracting periodic behavioral patterns from streaming mobility data, which fulfills the above-mentioned requirements. Our experimental results on both synthetic and real datasets confirm the superiority of our method in comparison with the existing techniques.

3.1 Introduction

Periodicity is an important characteristic of humans' and animals' activities. Animal's yearly migration, as well as, weekly work pattern of humans are examples of periodic behavioral patterns. Knowledge about activity periodicity is required by various applications. For example, ecologists are interested in knowing the periodic migration pattern of animals and how activities in vicinity of their living terrain cause abnormality in this behavior [160]. In humanitarian studies, it is interesting to identify interruptions in periodic routines by major life events or daily hassles, as this identification helps in understanding stress-induced changes in daily behavior of people [161]. Identification of such abnormalities in human behavior can be useful in designing solutions which alleviate the effect of such stresses (as used in participatory sensing-healthcare systems [7]). Apart from uncertainties associated with mobility data (such as noise and missing samples), which make

[4] This chapter is partly based on:

[159] M. Baratchi, N. Meratnia, and P. J. M. Havinga, "Recognition of periodic behavioral patterns from streaming mobility data," in *proceedings of the 10th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, Mobiquitous 2013*, Tokyo, Japan, 2013.

mining periodic patterns challenging, online extraction of patterns from streaming mobility data is difficult due to availability of limited processing and memory resources. The problem of identification of periodic behavioral patterns has been studied previously [162]. What distinguishes the work presented in this chapter from the existing research, however, is its focus on identification of periodic patterns from streaming mobility data through a light, and accurate technique.

3.1.1 Contributions

Our automatic pattern recognition method requires limited storage and processing capability and is able to detect periodic patterns upon arrival of every new mobility measurement. Our major contributions, in proposing such pattern recognition system, are:

- Accurate discovery of periods of repetitive patterns from streaming mobility data
- Extraction of periodic patterns with bounded memory requirement
- Performance evaluation using both synthetic and real datasets

The rest of this chapter is organized as follows. Related work is presented in Section 3.2. In Section 3.3, we will define the problem of finding periodic patterns from streaming mobility data. Our methodology is described in detail in Section 3.4. Section 3.5 and 3.6 present performance evaluation, and case studies with real datasets, respectively. Section 3.7 summarizes this chapter.

3.2 Related work

Existing solutions for pattern mining from mobility data can be divided into solutions addressing either *frequent* [163-167] or *periodic* pattern mining [162, 168, 169]. The former techniques focus on the "*number of times*" a pattern is repeated in limited duration (more representing the importance of a behavior in the number of time it repeats), while the latter focus on both the *"number of times"* a pattern is repeated and *"the temporal trend*" by which it is repeated. Considering the extra temporal trend, periodic patterns can provide extra information about the behavior of mobile entity.

Frequent pattern mining: Association rule mining [170] has been popularly used for extracting frequent trajectory patterns [163-167]. The general approach taken by this technique is to use a support-based mechanism to find the longest frequent trajectory pattern. Support-based mechanisms focus on the number of occurrences of patterns. The main drawback of exiting frequent pattern mining techniques is that the longest frequent pattern cannot completely and accurately describe the normal behavior. Specifically, these techniques [163-167] fail to detect behaviors

that do not occur frequently but they happen with higher prior expectation at a certain period.

Periodic pattern mining: There are a number of papers in the domain of time series analysis considering different questions regarding periodicity [171], such as asynchronous periodic patterns [172], and partial periodic patterns [173] of time series data. Recently, mining periodic patterns from mobility data has also received attention [162, 168, 169]. Use of signal processing techniques such as Fourier and wavelet transform was proposed in [174, 175]. As shown in [162], such forms of signal processing approaches perform weakly in presence of noise which make them inapplicable on raw mobility data. The authors of [162] proposed an automatic periodicity detection mechanism to find the periodic behaviors. They further extended their work for extracting periodicity from incomplete observations in [168]. They proposed a probabilistic measure for identifying periodicities in sparse mobility data. Their probabilistic measure is applied on data from visit to stay-points where the mobile entity spends a considerable amount of time. Similar to [168] we are interested in detection of periodic patterns from incomplete data. However, there are two main differences between these two techniques. Firstly, detection of periodic behavior in [168] is based on stay-points. Therefore, it is needed that the regions of interest are extracted beforehand. This requires a preprocessing phase, which is not needed by our technique, as we work with raw GPS measurements. Secondly, method of [168] is not designed for streaming data and consumes considerable amount of memory. Our method, on the other hand, has low resource consumption and complexity, which makes it applicable in streaming settings.

3.3 Problem Definition

In this section, we clearly define the problem of finding periodic patterns from streaming mobility data. We first start by providing a number of definitions:

Definition 3.1: *A trajectory $L_1, L_2, \dots, L_N$ is composed of a sequence of points denoted by $L_i = (x_i, y_i, t_i)$, where (x_i, y_i) represents a spatial coordinate and t_i is a time-stamp.*

Definition 3.2: *A period T is a time frame composed of T number of equally-sized segments denoted by $seg_{1..T}^{T}$.*

Definition 3.3: *A spatial neighborhood $sn_{(x_i,y_i)}$ is a set of location points that fall within the radius r of (x_i, y_i).*

Definition 3.4: *A spatial neighborhood is visited periodically in a period T, if the probability P_t^T of being in this neighborhood in a seg_t^T of period T is more than a threshold in all or a fraction of the observation time.*

Problem definition: Having limited memory available, we are interested in mining the last periodic pattern followed in data stream $(L_1 \dots L_i)$ in form of $<T, [(P_1^T, SN_1^T), \dots, (P_T^T, SN_T^T)] >$, where T is the temporal period and SN_t^T is a spatial neighborhood $sn_{(x_j,y_j)}$ which is expected to be visited periodically in seg_t^T with a period P_t^T.

3.4 Methodology to find periodic patterns (StPPattern)

Our method to find periodic patterns from streaming mobility data is composed of the following three stages (as shown in Figure 3-1):

- Measuring the self-similarity of the streaming data in different lags (described in Section 3.4.1);
- Discovery of the periods of repetition from the self-similarity graph (described in Section 3.4.2);
- Extracting periodic patterns (described in Section 3.4.3).

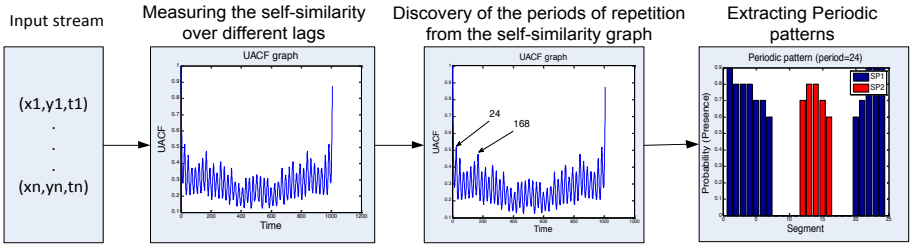


Figure 3-1 The framework for finding periodic patterns from streaming mobility data (StPPattern)

3.4.1 Measuring self-similarity of the mobility data in different lags

Behavioral patterns can have different periodicities (e.g. daily, weekly, monthly, and yearly). Therefore, it is important to first identify the period of repetition (day, week, or year) of visits to a certain spatial neighborhood. One of the most commonly used methods for identifying these periods is the circular Auto-Correlation Function (ACF) [176]. ACF measures the similarity of a time-series to itself in different lags (delay). ACF of a time series ts, of size N over lags $\tau \in \{1 \dots N\}$ is computed as follows:

$$ACF_N(\tau) = \sum_{i=1}^{N} ts(i).ts(i+\tau) \tag{3.1}$$

In its original form, ACF is not applicable to mobility data because the GPS data is often sparsely measured and mixed with noise (due to difficulties such as cloud cover, or device malfunction) while ACF requires the data to be uniformly sampled.

In order to measure the self-similarity from GPS measurements, we propose an optimization to the original ACF. Assuming that we denote missing samples as invalid and the rest as valid, we calculate the Uncertain circular Auto-Correlation Function ($UACF$) for a set of mobility data ($L_1 \dots L_N$) using Eq. (3.2):

$$UACF_N(\tau) = \frac{1}{v^{\tau}_{1..N}} \sum_{i=1}^{N} \Psi_{i,i+\tau} \tag{3.2}$$

Where $\Psi_{i,i+\tau}$ is equal to 1 when the Euclidean distance between a valid pair L_i and $L_{i+\tau}$ ($\text{dist}(L_i, L_{i+\tau})$) is less than a distance threshold r, and $v^{\tau}_{1..N}$ is the number of pairs $(i, i+\tau)$ in which both L_i, and $L_{i+\tau}$ are valid. Computing $UACF$ in this way will help us measure the self-similarity of GPS data only in an offline fashion when the entire mobility data is available. In the next section, we optimize the memory requirements of $UACF$ (Eq. (3.2)) and enable it to measure self-similarity over different lags upon arrival of each mobility data measurement.

3.4.1.1 Measuring self-similarity in streaming setting (online)

Finding periodic behavioral patterns in streaming setting helps reducing the data transmission and storage (as not the raw data but only the patterns or whether the entity conforms to the pattern can be transmitted or stored). Computing $UACF$ requires the entire data to be kept in memory. Therefore, its memory requirement is $O(N)$ (N is the number of measurements). Ubiquitous location-aware sensing devices have limited resources (both in terms of memory and power). Therefore, storing the entire data (especially, in case of high frequency sampled dataset) for a long period of time or its transmission to a central server for further analysis is neither practical nor is it possible. This motivates us to lower down the memory requirements. To do so, we need to calculate the $UACF$ in such a way that upon arrival of each new GPS measurement L_N, we can measure self-similarity over lags $\{\tau \,|N \bmod \tau = 0\}$. We claim that it is possible to reduce the memory requirement from $O(N)$ to $O(T_{max})$, by having an estimation of the maximum period being followed in data ($T_{max} \ll N$). (Since $N \bmod \tau = 0$, in what follows, we use $n\tau$ instead of N).

Theorem. Suppose that $L_1 L_2 \dots$ represent the stream of mobility data. We can compute $\{UACF_{n\tau}(\tau) | \ \tau \leq T_{max}\}$ for each $\{n > 3\}$ of this stream by having $O(T_{max})$ memory.

Proof. In order to prove the above theorem we first prove that we can re-compute Eq. (3.2) in an alternative way. Consequently, we prove that in its new form, the

memory requirement of computing $UACF$ is bounded by $6{\times}T_{max}$. Therefore, we will first prove through mathematical induction that for each $(n > 3)$, $UACF_{n\tau}(\tau)$ can be computed as follows:

$$\mathrm{U}ACF_{n\tau}(\tau) = \frac{1}{v^{\tau}_{1..n\tau}}\left(v^{\tau}_{1..(n-1)\tau}\left(UACF_{(n-1)\tau}(\tau)\right) - \sum_{i=1}^{\tau}\Psi_{i,(n-2)\tau+i} + \sum_{i=1}^{\tau}\Psi_{(n-2)\tau+i,(n-1)\tau+i} + \sum_{i=1}^{\tau}\Psi_{(n-1)\tau+i,i}\right) \tag{3.3}$$

Base Step. The base step is to check the validity of the above equation for $n = 4$. For $n = 4$ computing $UACF_{4\tau}(\tau)$ by Eq. (3.2) results in Eq. (3.4) and computing this value by Eq. (3.3) will result in Eq. (3.5) (please note that due to circular shift operation ($\sum_{i=1}^{\tau}\Psi_{2\tau+i,3\tau+i} = \sum_{i=1}^{\tau}\Psi_{2\tau+i,i}$):

$$UACF_{4\tau}(\tau) = \frac{1}{v^{\tau}_{1..4\tau}}\sum_{i=1}^{4\tau}\Psi_{i,i+\tau} = \frac{1}{v^{\tau}_{1..4\tau}}\left(\sum_{i=1}^{\tau}\Psi_{i,\tau+i} + \sum_{i=1}^{\tau}\Psi_{\tau+i,2\tau+i} + \sum_{i=1}^{\tau}\Psi_{2\tau+i,3\tau+i} + \sum_{i=1}^{\tau}\Psi_{3\tau+i,i}\right) \tag{3.4}$$

$$UACF_{4\tau}(\tau) = \frac{1}{v^{\tau}_{1..4\tau}}\left(\sum_{i=1}^{\tau}\Psi_{i,2\tau+i} + \sum_{i=1}^{\tau}\Psi_{2\tau+i,3\tau+i} + \sum_{i=1}^{\tau}\Psi_{3\tau+i,i}\right) \tag{3.5}$$

We replace $UACF_{3\tau}(\tau)$ in Eq. (3.5) to see if it equals to Eq. (3.4). Using Eq. (3.2) we will have:

$$UACF_{3\tau}(\tau) = \frac{1}{v^{\tau}_{1..3\tau}}\sum_{i=1}^{3\tau}\Psi_{i,i+\tau} = \frac{1}{v^{\tau}_{1..3\tau}}\left(\sum_{i=1}^{\tau}\Psi_{i,\tau+i} + \sum_{i=1}^{\tau}\Psi_{\tau+i,2\tau+i} + \sum_{i=1}^{\tau}\Psi_{2\tau+i,i}\right) \tag{3.6}$$

By replacing $UACF_{3\tau}(\tau)$ in Eq. (3.5) with Eq. (3.6) we achieve Eq. (3.4) as:

$$\begin{aligned} UACF_{4\tau}(\tau) = {} & \frac{1}{v^{\tau}_{1..4\tau}}\left(v^{\tau}_{1..3\tau}.\left(\frac{1}{v^{\tau}_{1..3\tau}}\right)\left(\sum_{i=1}^{\tau}\Psi_{i,\tau+i} + \sum_{i=1}^{\tau}\Psi_{\tau+i,2\tau+i} + \sum_{i=1}^{\tau}\Psi_{2\tau+i,i}\right)\right. \\ & \left. - \sum_{i=1}^{\tau}\Psi_{i,2\tau+i}) + \sum_{i=1}^{\tau}\Psi_{2\tau+i,3\tau+i} + \sum_{i=1}^{\tau}\Psi_{3\tau+i,i}\right) \\ = {} & \frac{1}{v^{\tau}_{1..4\tau}}\left(\sum_{i=1}^{\tau}\Psi_{i,\tau+i} + \sum_{i=1}^{\tau}\Psi_{\tau+i,2\tau+i} + \sum_{i=1}^{\tau}\Psi_{2\tau+i,3\tau+i} + \sum_{i=1}^{\tau}\Psi_{3\tau+i,i}\right) \end{aligned} \tag{3.7}$$

Induction step. Assuming that $\{k \in N | k > 3\}$ is given and Eq. (3.3) is true for $n = k$. Then we can prove that the Eq. (3.3) is valid for $n = k + 1$:

$$
\begin{aligned}
UACF_{(k+1)\tau}(\tau) &= \frac{1}{v^{\tau}_{1..(k+1)\tau}} \sum_{i=1}^{(k+1)\tau} \Psi_{i,i+\tau} \\
&= \frac{1}{v^{\tau}_{1..(k+1)\tau}} \left(\sum_{i=1}^{\tau} \Psi_{i,\tau+i} + \cdots + \sum_{i=1}^{\tau} \Psi_{((k+1)-3)\tau+i,((k+1)-2)\tau+i} \right. \\
&\quad \left. + \sum_{i=1}^{\tau} \Psi_{((k+1)-2)\tau+i,((k+1)-1)\tau+i} \right) \\
&= \frac{1}{v^{\tau}_{1..(k+1)\tau}} \left(\left(\frac{v^{\tau}_{1..k\tau}}{v^{\tau}_{1..k\tau}} \right) \left(\sum_{i=1}^{\tau} \Psi_{i,\tau+i} + \cdots + \sum_{i=1}^{\tau} \Psi_{(k-2)\tau+i,(k-1)\tau+i} \right) \right. \\
&\quad \left. + \sum_{i=1}^{\tau} \Psi_{((k+1)-2)\tau,((k+1)-1)\tau+i} + \sum_{i=1}^{\tau} \Psi_{((k+1)-1)\tau+i,i} \right) \\
&= \frac{1}{v^{\tau}_{1..(k+1)\tau}} \left((v^{\tau}_{1..k\tau}) \left(\frac{1}{v^{\tau}_{1..k\tau}} \right) \left(\sum_{i=1}^{\tau} \Psi_{i,\tau+i} + \cdots \right. \right. \\
&\quad \left. + \sum_{i=1}^{\tau} \Psi_{(k-2)\tau+i,(k-1)\tau+i} + \sum_{i=1}^{\tau} \Psi_{(k-1)\tau+i,i} \right) - \sum_{i=1}^{\tau} \Psi_{(k-1)\tau+i,i} \\
&\quad \left. + \sum_{i=1}^{\tau} \Psi_{((k+1)-2)\tau,((k+1)-1)\tau+i} + \sum_{i=1}^{\tau} \Psi_{((k+1)-1)\tau+i,i} \right) \\
&= \frac{1}{v^{\tau}_{1..(k+1)\tau}} \left(v^{\tau}_{1..k\tau} . (\mathrm{U}ACF_{k\tau}(\tau)) - \sum_{i=1}^{\tau} \Psi_{((k+1)-2)\tau+i,i} \right. \\
&\quad \left. + \sum_{i=1}^{\tau} \Psi_{((k+1)-2)\tau,((k+1)-1)\tau+i} + \sum_{i=1}^{\tau} \Psi_{((k+1)-1)\tau+i,i} \right)
\end{aligned}
\tag{3.8}
$$

Now, we prove that we can calculate Eq. (3.3) with bounded memory. In this equation, $\sum_{i=1}^{\tau} \Psi_{(n-1)\tau+i,i}$ is calculated from $L_{1\dots\tau}$ and $L_{(n-1)\tau+1\dots n\tau}$. $\sum_{i=1}^{\tau} \Psi_{(n-2)\tau+i,(n-1)\tau+i}$ is calculated from $L_{(n-2)\tau+1\dots n\tau}$. $UACF_{(n-1)\tau}(\tau)$ and $\left(\sum_{i=1}^{\tau} \Psi_{(n-2)\tau+i,i}\right)$ are single values computed in the previous round. Using induction, it is straightforward to prove that we can also compute $v^{\tau}_{1\dots n\tau}$ from $v^{\tau}_{1\dots(n-1)\tau}$ through $(v^{\tau}_{1\dots n\tau} = v^{\tau}_{1\dots(n-1)\tau} - v^{\tau}_{(n-2)\tau\dots\tau} + v^{\tau}_{(n-2)\tau\dots n\tau})$, where $v^{\tau}_{(n-2)\tau\dots\tau}, v^{\tau}_{(n-2)\tau\dots n\tau}$ are computed from $L_{1\dots\tau}$ and $L_{(n-1)\tau+1\dots n\tau}$. We know that $(\tau \le T_{max})$ so $(L_{1\dots\tau} \in L_{1\dots T_{max}})$ and $(L_{(n-2)\tau+1\dots n\tau} \in L_{(n\tau-2T_{max}+1)\dots n\tau})$. Therefore, if we have $L_{1\dots T_{max}}$, $L_{(n\tau-2T_{max}+1)\dots n\tau}$ (which require on maximum $3T_{max}$memory) and$\{v^{\tau}_{1\dots n\tau}, UACF_{(n-1)\tau}(\tau), \sum_{i=1}^{\tau} \Psi_{i,(n-2)\tau+i} | \tau < T_{max}\}$(also with maximum $3T_{max}$memory) in memory we can compute $UACF_{N=n\tau}(\tau)$ for any τ. Thereby, instead of keeping N measurements in memory we only need to keep $6\times T_{max}$ $(T_{max} \ll N)$ values and the rest of data can be destroyed.

3.4.2 Discovery of periods of repetition

If there is a single period of repetition in a time-series, the self-similarity graph (with both *ACF* and *UACF*) will show a peak in that period and its entire integer multiples. For instance, if there is a pattern repeated with period of 24 hours, the peaks will appear at 24, 48, 72, and so on. In order to extract periods of repetition from the self-similarity graph, normally the first highest peak is chosen. We cannot ignore the fact that there may exist multiple periodic patterns in mobility data. Therefore, it is advantageous to be able to extract all periodic patterns rather than the ones with the first highest peak. To clarify the case, in which multiple periodic patterns exist, we provide an example.

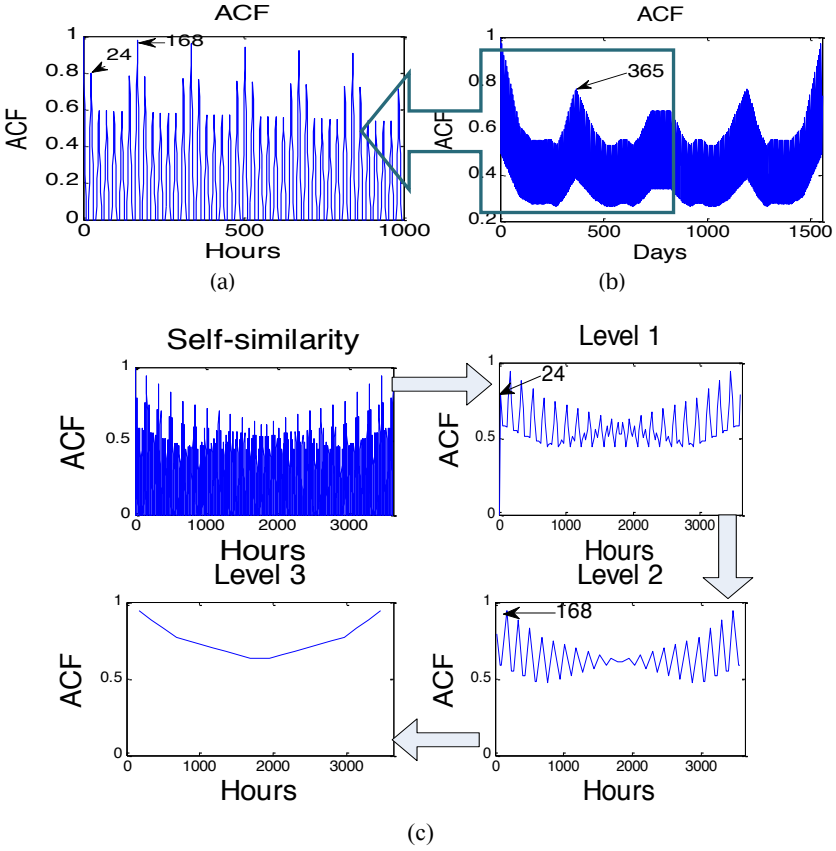


Figure 3-2 (a) *ACF* self-similarity graph on the presence sequence of Bob on visiting school for the first 1000 hours of 4 years (τ=1 hrs), (b)The result of performing *ACF* on the presence sequence data of Bob on visiting his school (τ=24 hrs), (c) Removing peaks level by level to extract periods of repetition (Algorithm 3.1)

Let us consider Bob, a student, who goes to school every weekday during the study year and stops going to school during summer. From one perspective, this behavior is periodic over a year (9 months going to school and 3 months of holiday). From another view, we can also observe some other periods of repetition in this behavior (24 hours, 7 days) as Bob goes to school every weekday and stops going to school on weekends. If we build a binary presence sequence for this activity of Bob for four years by placing 1 at each time stamp when Bob is present at school and 0 at other times, the self-similarity graph by computing ACF on this sequence will look like Figure 3-2.a and Figure 3-2.b.

As seen in self-similarity graph in Figure 3-2.a and Figure 3-2.b, there are multiple hierarchically ordered valleys and hills. The peaks with the highest ACF values are the ones, which belong to the multiples of longer periods (in this example 365 days). The lower hills, on the other hand, belong to multiples of shorter periods (24 and 168). We can see in Figure 3-2.c that, if we iteratively extract peaks of the self-similarity graph, such periods can be found by choosing the first peak in each iteration. This will enable us to define periods of repetition as:

Definition 3.5: *Time lags $T_1 \dots T_n$ are the periods of repetition in a data stream if (i) the self-similarity graph has a local maximum in lags $T_1 \dots T_n$, and (ii) T_i is the first peak among peaks of level i-1, which is repeated in integer multiplies ($2T_i, 3T_i, \dots$).*

Our procedure of extracting the periods of repetition is presented in Algorithm 3.1 (PeriodExtract).

Algorithm 3.1 (PeriodExtract)

INPUT: $UACF_N(1 \dots N)$(self-similarity graph)
OUTPUT: T (set of periods)

Find first level peaks, $Peaklevel(1)$ from $UACF(1 \dots N)$; //Finding first level local maximums
set $i = 1$;// Counter of peal levels
Repeat while $Peaklevel(i)$ is not empty
    Find $Peaklevel(i+1)$ among $Peaklevel(i)$;// As depicted in Figure 3.2
    set $i = i + 1$;
End while
For each $(j < i)$
    Set period $T(j)$ to the first peak in $Peaklevel(j)$ which is repeated in integer multiplies;
End for

3.4.3 Extracting periodic patterns in streaming setting

Successful discovery and extraction of periods of repetition only informs us of periodic visits to some spatial neighborhoods. This, however, does not indicate which spatial neighborhoods and when (in which segment of the period) they have been visited. Considering that the random existence of a mobile entity in a spatial neighborhood $sn_{(x_j,y_j)}$ at seg_t^T of a discovered period T follows a Bernouli distribution [177] (being in $sn_{(x_j,y_j)}$ (1), not being in $sn_{(x_j,y_j)}$ (0)), the probability that this entity appears in $sn_{(x_j,y_j)}$ at seg_t^T randomly is ½. If this probability is more than ½, it shows that the mobile entity has a tendency to appear in that $sn_{(x_j,y_j)}$ and its visit conforms to a periodic pattern. Therefore, in order to find the periodic patterns we need to find spatial neighborhoods, which have been visited with a probability more than ½ in each segment of the discovered period of repetition.

Algorithm 3.2 (PeriodicPatternExtract)

INPUT: L_N**(***data point***),** Buffer, $PL^{T=1\ldots TMax}=[P_{i..T}^T, V_{i..T}^T, SN_{i..T}^T]$, T_{max}, r**(radius)**
OUTPUT: Buffer, $PL^{T=1\ldots TMax}=[P_{i..T}^T, V_{i..T}^T, SN_{i..T}^T]$, $PPatterns_{1\ldots T_{max}}$

1: Add L_N to the end of the Buffer and remove a point from the beginning of
2: Buffer;
3: Update $UACF_N(\tau)$where N mod $\tau = 0$;// Eq. (3.6)
4: Find periods of repetition $T_{1\ldots k}$ from self-similarity graph
5: $UACF$ $(1 \ldots T_{max})$;// Algorithm (3.1)
6: **For each** period T_i in periods $T_{1\ldots k}$
7: $t = N \bmod T_i$
8: **If** (dist $(SN_t^T, L_N)<2r$)
9: $P_t^{T_i} = P_t^{T_i}+1$, $SN_t^{T_i} = (P_t^{T_i}.SN_t^{T_i} + L_n)/(P_t^{T_i}+1)$;
10: **Else if** $(\frac{P_t^{T_i}}{V_t^{T_i}} < 1/2)$, $SN_t^{T_i} = L_n$, $P_t^{T_i} = 1, V_t^{T_i}=0$;
11: $V_t^{T_i} = V_t^{T_i} + 1$;
12: **End if**
13: $PPattern_{T_i}=\{SN_t^{T_i} \mid P_t^{T_i} > 1\}$
14: **End for**

Algorithm 3.2 (PeriodicPatternExtract) summarizes how we can extract both temporary and permanently periodic behaviors from streaming data. The algorithm proceeds as follows. Firstly, we use $UACF$ to extract the periods. Next, for each discovered period of repetition T_i, we update the entries of a list of size T_i (referred to as PL^{T_i}, $PL^{T_i}=[(P_1^{T_i}, V_1^{T_i}, SN_1^{T_i}), \ldots, (P_{T_i}^{T_i}, V_{T_i}^{T_i}, SN_{T_i}^{T_i})]$). For each spatial neighborhood $SN_i^{T_i}$, $P_i^{T_i}$ denotes the number of presences in $SN_i^{T_i}$ and $V_i^{T_i}$ represents the number of

valid observations $V_i^{T_i}$ in segment $seg_i^{T_i}$. In each timestamp entities of PL^{T_i} lists get updated. Each measurement $\{L_N | N \bmod T_i = t\}$ will be compared with the value of $SN_t^{T_i}$of PL^{T_i} list. In case the measurement lies within $2r$ from $SN_t^{T_i}$, the value of $SN_t^{T_i}$ will be updated with the average of the previous $SN_t^{T_i}$ values and the new value L_N. The values of $P_t^{T_i}$and $V_t^{T_i}$ will also be updated correspondingly. Finally, the pattern composed of the value of spatial neighborhoods with a probability over ½ will be returned as periodic pattern and those spatial neighborhoods ($SN_t^{T_i}$) with a probability less than ½ will be removed.

3.5 Evaluation

3.5.1 Complexity analysis

In this section, we analyze the processing complexity and memory resources needed for extracting periodic patterns from streaming data (StPPattern) of size N by Algorithm 3.2 (PeriodicPatternExtract) assuming that the maximum repetitive period in the stream is less than T_{max}. We compare our method with the method proposed in [168] and with the original ACF [176]. It should be mentioned that ACF and [168] only measure self-similarity. Therefore, we only have to address their memory and processing power for this task. In our method, StPPattern, arrival of each new point, extracting repetition periods, and updating the PL lists have processing complexity of (T_{max}), $O(T_{max} log T_{max})$, and $O(T_{max}^2)$, respectively. As shown in Section 3.4.1.1, we reduced the memory requirements of measuring self-similarity to $O(T_{max})$ and discovery of the periods of repetition has memory complexity of $O(T_{max})$. In pattern extraction, we keep a list of size T (PL) for each period $(T < T_{max})$. Therefore, memory requirement of this task (PeriodicPatternExtract Algorithm) is $O({T_{max}}^2)$. The method proposed in [168] extracts periodicities from each region of interest rather than from the original data stream. In order to perform streaming period extraction, this method should be able to identify the regions of interest first, which is not needed by our technique. The regions of interest are not known beforehand. Therefore, to be able to compare our technique with [168], we simply assume that we compare each new GPS measurement with cells of a grid of size G. In this case, the processing complexity for this comparison will be $O(G)$. In order to measure the self-similarity, method of [168] requires having all the previous data points in memory. As new data arrives, it needs to update probability of presence in each segment of each period. Then it measures the self-similarity for each possible period, with a complexity of $O(T_{max}N)$. This task should be performed C number of times (C is a constant value) in order to normalize the data. Therefore, the total computational complexity is $O(CNT_{max}) + O(G)$ and memory requirements will be $O(N)$. Complexity of ACF using Eq. (3.1) is $O(N^2)$ and it also requires the whole data to be stored in

memory. Table 3-1 summarizes the memory and processing complexity of these three techniques. As seen, concerning the required memory and computational resources, StPPattern is best suited for streaming settings and resource restricted devices.

Method	Processing			Memory		
	Measuring self-similarity	Period extraction	Pattern extraction	Measuring self-similarity	Period extraction	Pattern extraction
StPPattern	$O(T_{max})$	$O(T_{max} log T_{max})$	$O({T_{max}}^2)$	$O(T_{max})$	$O(T_{max})$	$O({T_{max}}^2)$
[168]	$O(G)+$ $O(GNT_{max})$	-	-	$O(N)$	-	-
ACF [176]	$O(N^2)$	-	-	$O(N)$	-	-

Table 3-1 Complexity comparison

3.5.2 Performance in presence of uncertainties

3.5.2.1 Description of synthetic dataset

In this section we validate the performance of Algorithm 3.2 (PeriodicPatternExtract) using a synthetic dataset to test its sensitivity under parameters, which cause imperfections in mobility data. For this purpose, we implemented a mobile object sequence generator to produce a sequence representing a person's periodic movement in N number of days. This periodic sequence is in form of a test sequence ($test_i = \{(x_i, y_i) | i \in [1, N \times 24]\}$), where each index represents a spatial neighborhood in which a person is between $[(i-1)\ mod\ 24, i\ mod\ 24]$ in the $(\frac{i}{24}+1)^{th}$ day. Ten spatial neighborhoods are defined, each composed of two-dimensional points lying within radius r from a predefined center. We consider two of these spatial neighborhoods (representing home and office) being periodically visited (daily, and weekly) in specific intervals. For workdays, the interval 10:00-18:00 is chosen for "being at work" and 20:00-8:00 for "being at home". On weekends, the interval between 01:00-24:00 is chosen for "being at home". Each of these intervals is subject to a random event with probability of μ and is normal otherwise. In normal intervals with defined start (t_{start}) and end (t_{end}), the event of "visit" (being at home or office) starts somewhere between ($t_{start} \pm \sigma_1$) and ends around ($t_{end} \pm \sigma_2$). The behavior in abnormal intervals is randomly chosen from the other 9 spatial neighborhoods with a random start-time and random duration. Such abnormal intervals can represent different un-periodic events such as absence at work, working overtime,

or visit to places such as cinemas, shops, etc. After defining the normally and abnormally visited places (spatial neighborhoods) for each day, we add trajectories between them, each with different duration. This can represent different modes of transport, (for instance, car, or bike). The effect of missing samples was tested by removing data from the random indexes (both (x, y)) with probability of α. In order to add noise, we formed a randomly permuted array of data between the maximum and minimum longitude and latitudes in selected spatial neighborhoods. Next, we randomly picked samples with probability of β and replaced them with the values in the random array. The parameters used to form the test sequence are: radius of spatial neighborhood (r =100 meters), number of periodic repetition (N =100), missing samples (α =0-50%), noise (β =0-50%), standard deviation of start/end-time (σ_1, σ_2 = 2), and probability of random events (μ =0-50%).

3.5.2.2 Performance evaluation with the synthetic dataset

The synthetic dataset generated by movement generator follows two periods of repetition of 24, and 168 hours (corresponding to a day and a week). In this section, we evaluate Algorithm 3.1 (PeriodExtract) in terms of its success in correctly extracting these two periods using ACF and $UACF$ self-similarity graphs (method of [168] is not applicable on raw data). We calculate self-similarity in different lags by ACF on latitude (lat), longitude ($long$) and their root mean square ($RMS = \sqrt{lat^2 + long^2}$). Running these experiments 100 times, we test the effect of uncertainty parameters, noise, missing samples, and random events on detection of correct periods. In each experiment the samples to be changed based on these parameters were selected randomly. Figure 3-3 (a-c) compare $UACF$ and ACF ($Lat, Long, RMS$) in correctly identifying the period of 24 form the synthetic dataset. These techniques are compared in finding the period of 168 in Figure 3-3 (d-f). Each graph represents the performance of these algorithms under a certain percentages of these uncertainty parameters (x axis label). Generally, looking at these graphs it is inferred that $UACF$ clearly outperforms ACF in presence of noise, missing samples, and random events. Even when these uncertainty parameters reach 50%, $UACF$ still can find a high percentage of correct periods. This outstanding performance of $UACF$, compared to ACF, is achieved by the measuring self-similarity only for the points that fall within a spatial neighborhood ($\Psi_{i,i+\tau}$).

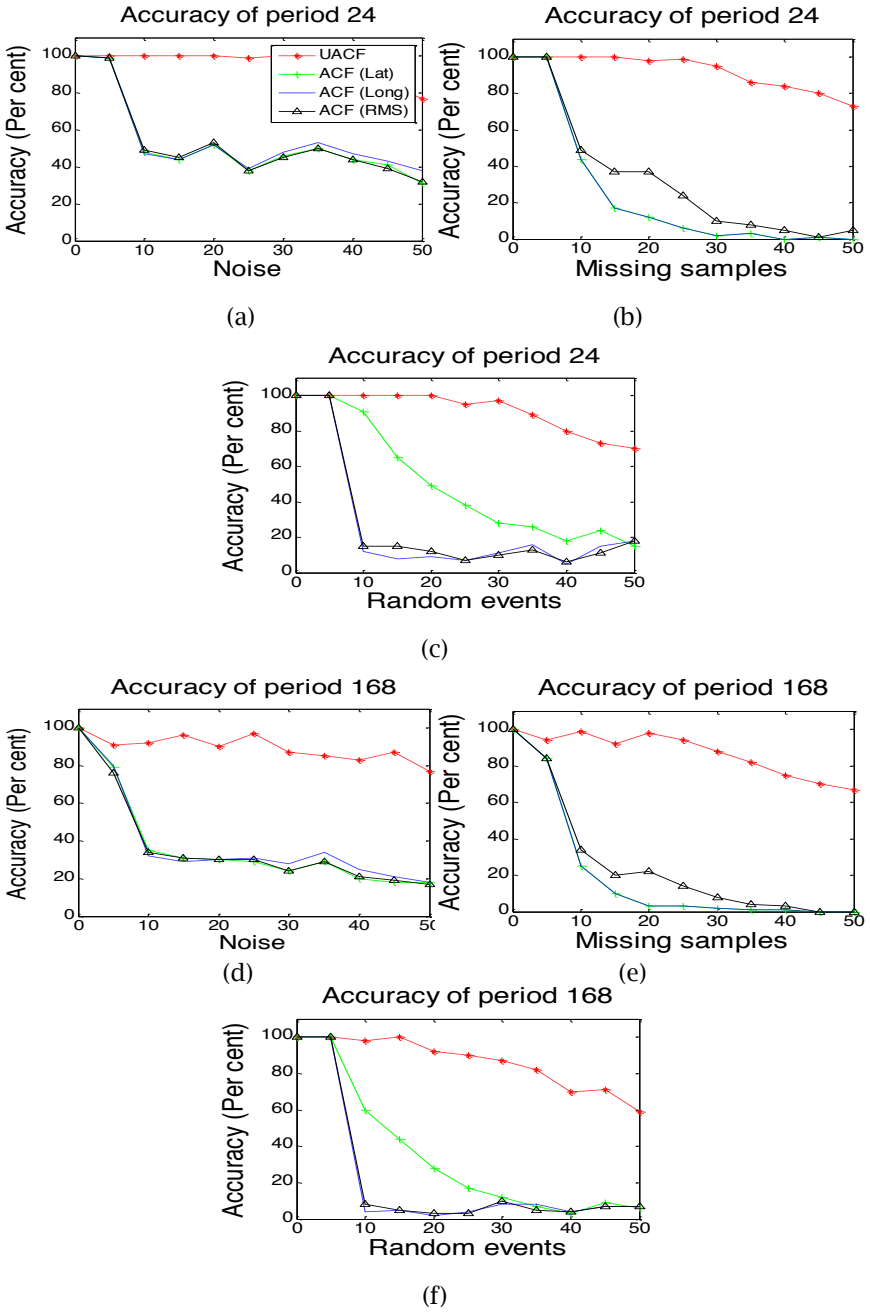


Figure 3-3 (a-f) Comparison of the accuracy of Algorithm 3.1 in extracting periods of repetition 24,168) using *UACF* and *ACF* in presence of noise, missing samples and random events

This way, *UACF* overcomes the effect of sparse, pattern-less, and noisy data. *ACF*, however, measures self-similarity by multiplication of changed samples with the unchanged ones, which follow a pattern. Therefore, noise and random events can clearly degrade the performance of *ACF* leaving no effect on that of *UACF*.

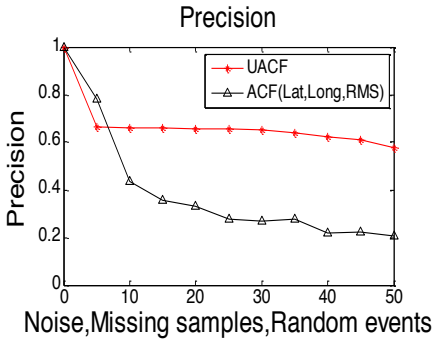


Figure 3-4 Average precision of Algorithm 3.1 in extracting periods of repetition

Accuracy, as shown in Figure 3-3, relates to how well algorithms extract the correct period from the mobility time series. The periodicity detection method might be accurate but not precise. Meaning that, other than extraction of correct periods, it may also extract wrong periods. An ideal self-similarity measure should offer both an accuracy and a precision measure close to 1. Figure 3-4 compares the precision computed by $\frac{P^+}{P^+ + P^-}$ where P^+ is the sum of correct prediction of two periods and P^- is the number of false alarms in all the previous experiments (an average for both periods of repetition and all uncertainty parameters). As Figure 3-4 shows, the overall precision of *UACF* is also higher than that of *ACF*. An increase in the amount of noise, missing samples, and random events, cause the precision of *ACF* to rapidly fall below 0.5. This shows that the number of false alarms exceed the correct prediction. *UACF* Algorithm, however, maintains its precision, which only decreases slightly as the uncertainties increase.

3.6 Case studies

Having proved the validity of *UACF* in finding periods of repetition, in this section, we describe our experiments with two of the previously mentioned datasets.

3.6.1 Case study using Dataset 1

We applied *UACF* and *ACF* (root sum of square) on Dataset 1 to measure self-similarity over different time lags (results shown in Figure 3-5.a-e, Figure 3-6.a-e). Algorithm 3.1 (PeriodExtract) was used to extract periods of repetition for two

entities in this dataset. For the first entity, we were able to extract the period of 24 hours using *UACF*, while no period was found using *ACF*. We noticed that it was not possible to extract the period of 168 as no data were available for weekends.

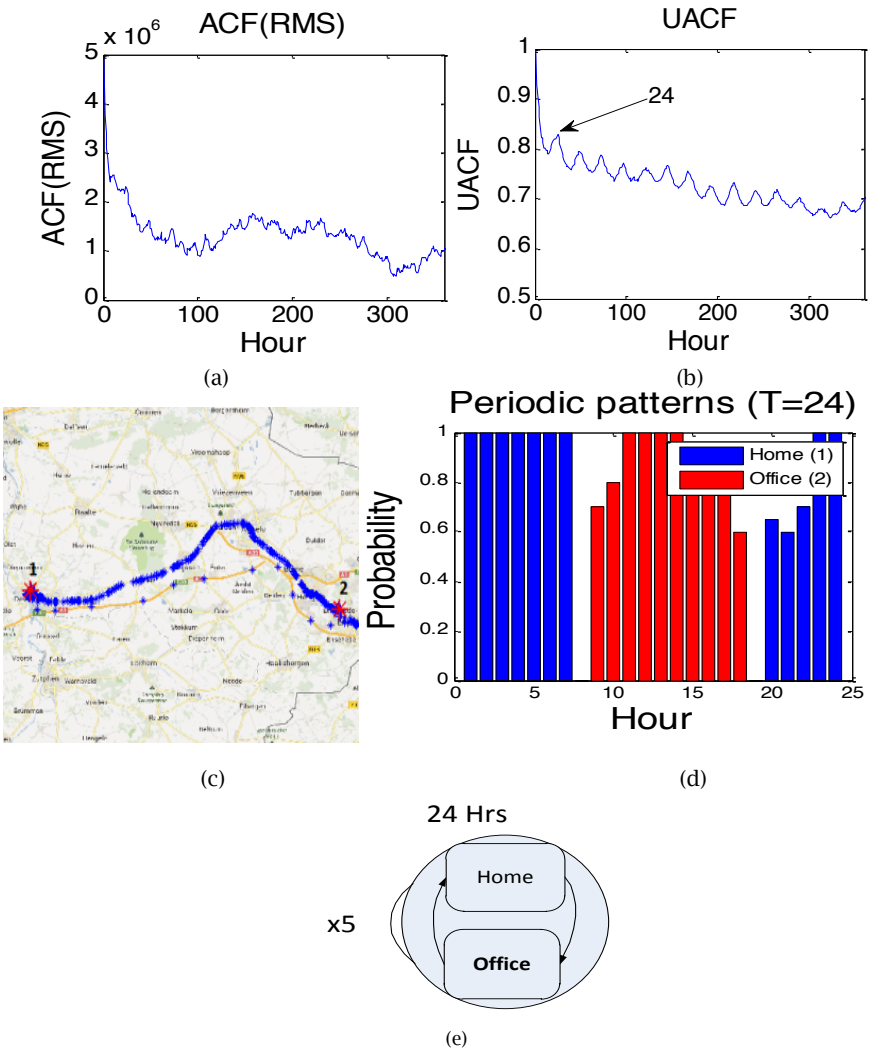


Figure 3-5 Extracting periodic behavior of the first entity in dataset 1. (a,b) extracting periods from self-similarity graph of real dataset using *ACF* and *UACF*, (c) mobility data stream (shown in blue) and identified periodically visited spatial neighborhood (shown in red), (d) periodic patterns extracted from algorithm 2, (e) state-diagram of periodic behavior

For the second entity, using *UACF* we were able to detect both periods of 24 and 168 hours, while with *ACF* only the period of 24 could be found. It can be seen from Figure 3-6.b-c, that the lag of 24 represents the first highest peak in *ACF* graph while there is no distinguishable peak afterwards. This huge difference in performance of *ACF* and *UACF* is due to long travels, which show themselves as random events. While the random events cannot degrade the performance of *UACF*, as shown in Section 3.5.2.2, they do considerably degrade the performance of *ACF*. The hierarchy of peaks is clearly distinguishable using *UACF*. Therefore, both periods were easily found using Algorithm 3.1 (PeriodExtract).

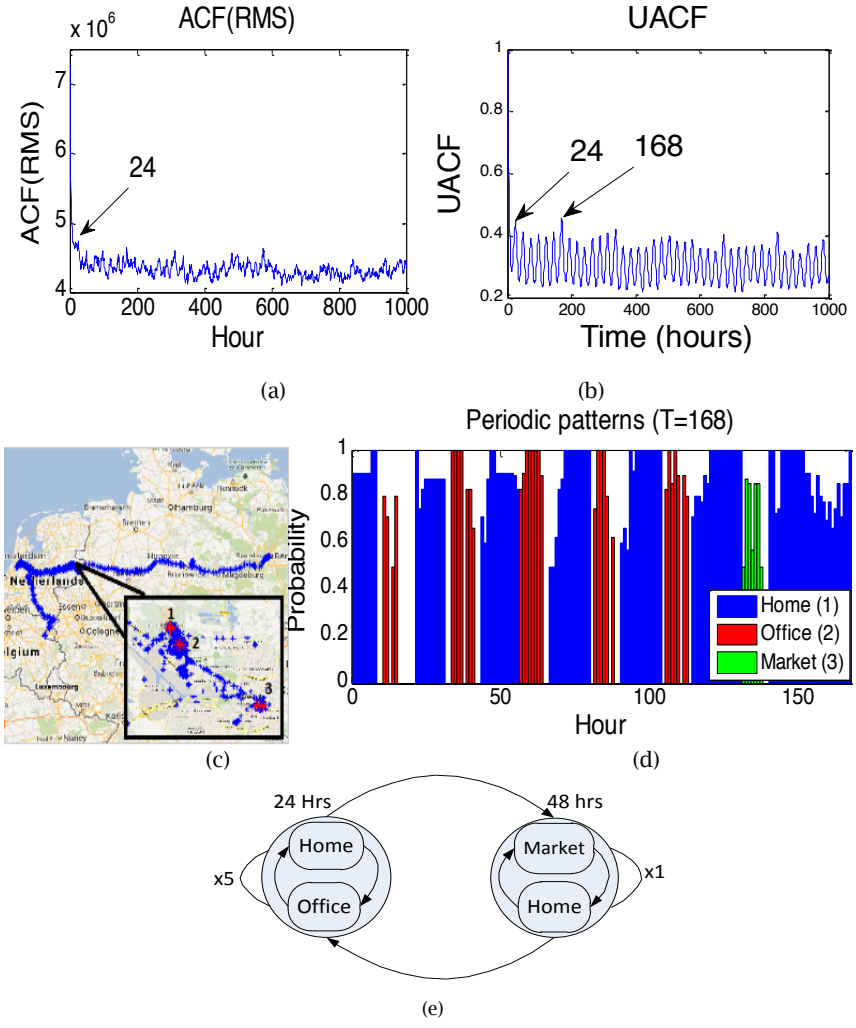


Figure 3-6 Extracting periodic behavior of the second entity in dataset 1. ((a-e) The same as Figure 3-5

After finding the spatial neighborhoods for each segment of the discovered periods using Algorithm 3.2 (PeriodicPatternExtract) we merged those which were lying within a spatial neighborhood. Using this algorithm we were able to find two spatial neighborhoods for the first entity (his home and office) (Figure 3-5.a) and three spatial neighborhoods for the second entity (home, office, and market) (Figure 3-6.a).

The histograms in Figure 3-5.d and Figure 3-6.d represent the probability of appearance in SP_i^T in segment seg_i^T of each of the larger discovered periods. The state diagrams on the right are drawn based on the histograms to represent the periodic pattern. As illustrated in the state diagrams, the periodic pattern of the first entity is composed of a loop between home and work. For the second entity, a periodic pattern of two loops is identified. These loops are repeated 5 times with the duration of 24 hours (Weekdays). Next, a new loop of 48 hours emerges, which is only followed once. Afterwards, the first loop is repeated again.

3.6.2 Case study using Dataset 2

In Dataset 2, the sampling schedule through which the dataset is acquired from Capricorn is fixed but uneven. This dataset was collected such that samples are acquired when the animal is known to be mostly active. Thus, there are 8 constant missing values in selected hours of day. To avoid being biased, we removed these constant missing samples from the time series, forming an evenly sampled time series. After plotting the self similarity measured by $UACF$ and ACF (with the spatial neighborhood of radius 500 meters) a dominant peak is observed, which is near 16 considering the 8 empty timestamps, this value can represent the 24 hours daily periodic pattern. For this dataset, both $UACF$ and ACF can find the period of 16 accurately. As opposed to the previous dataset, ACF is also applicable here. This is due to the limited movement range of the Capricorn. As seen in Figure 3-7, eventually by merging adjacent spatial neighborhoods, a small spatial neighborhood is found. The center of this spatial neighborhood with the radius of 500 meters is shown in Figure 3-7.c with red. The periodic pattern of visit to this spatial neighborhood is shown in Figure 3-7.d which corresponds to the last valid periodic behavior of this animal in special timestamps of the day. This means that, it is highly expected that this animal is present in the spatial neighborhood shown in Figure 3-7.c during the timestamps shown in Figure 3-7.d.

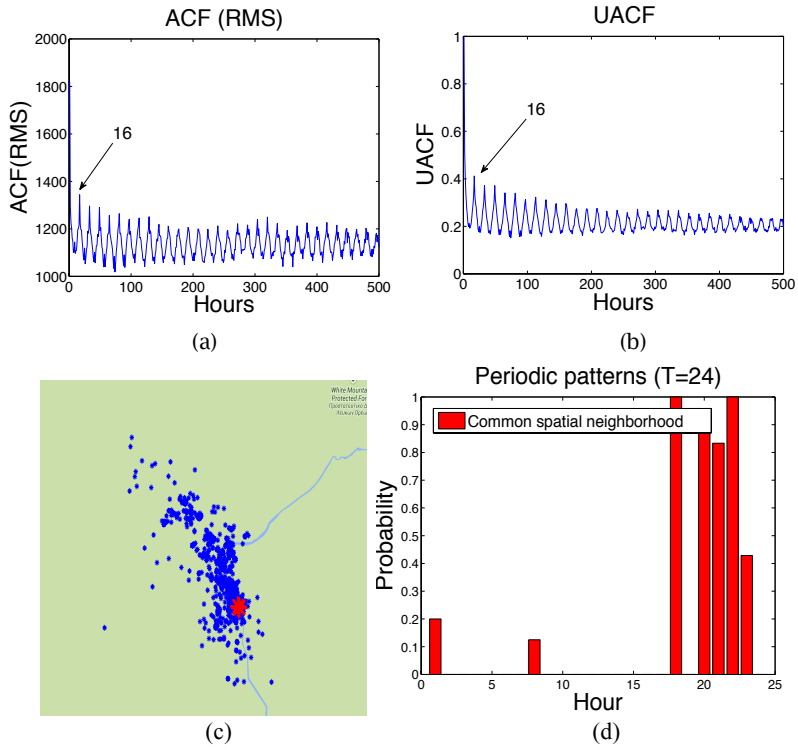


Figure 3-7 Extracting periodic behavior of the first entity in Dataset 2. (a-d) The same as Figure 3-5

3.7 Summary

In this chapter, we addressed the problem of extracting periodic behavioral patterns from streaming mobility data. Firstly, we proposed a self-similarity measure to identify periods of repetition from raw streaming mobility measurements. We proved that as opposed to the other methods used for extracting periods of repetition from mobility data, the memory requirement of this method is controllable and can be bound to the resources available. We empirically evaluated the performance of our method using a synthetic dataset under different controllable uncertainty parameters such as noise, missing samples, and random events. Results of our evaluations on the synthetic dataset shows that the self-similarity measure, which we proposed for identifying periods of repetition in mobility data, is strongly resistant to noise, missing samples, and random events. We further used the periods found based on this similarity measure to extract periodic patterns. Case studies with real datasets represent how

the result of period extraction can be used to visualize the periodic behavior of the mobile entity.

Chapter 4

Trajectory modeling[5]

A trajectory model can capture high-level semantics of the spatio-temporal mobility data. By reducing randomness, such model can make the trajectories more understandable. Thus, they can better be used in different applications such as future movement predictions or movement anomaly detection. In this chapter, we study the problem of modeling trajectories by looking for associations in consecutive mobility data. We address the problem of trajectory modeling using both deterministic and probabilistic approaches. In the first approach, we directly break down trajectories to find their smallest meaningful segments and then count their frequency of occurrence. In the second approach, we use generative state-space modeling techniques to probabilistically model trajectories.

4.1 Introduction

All applications and services, which use spatio-temporal mobility data, depend on availability of some knowledge about the behavior of mobile entities. Having such knowledge helps us predict future mobility patterns, as well as identify abnormal occurrences in the current mobility patterns. A detailed movement model, which identifies patterns of visit to frequently visited places, greatly contributes to acquiring this knowledge.

---

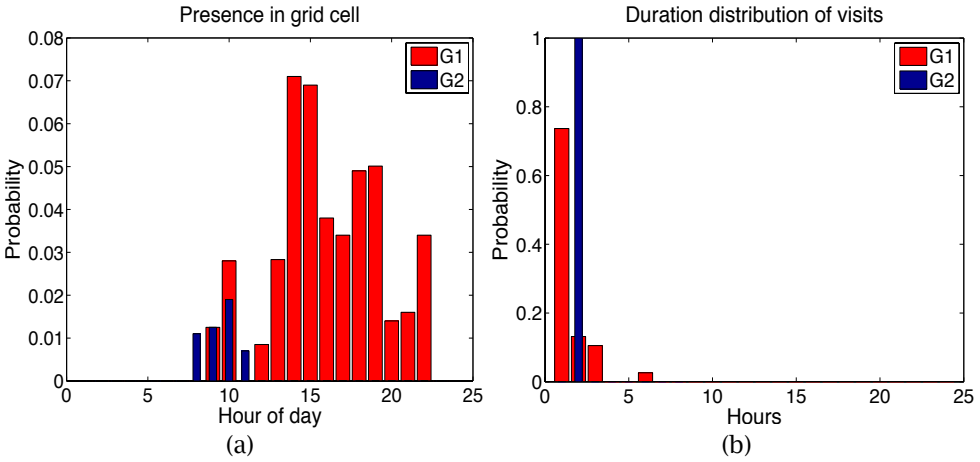
[5] This chapter is partially based on:

[167] M. Baratchi, N. Meratnia, and P. J. M. Havinga, "Finding frequently visited paths: dealing with the uncertainty of spatio-temporal mobility data," in *proceedings of the 2013 IEEE Eighth International Conference on Intellifent Sensor, Sensor Networks and Information Processing (ISSNIP'13)*, Melbourne, Australia, 2013, pp. 479-484.

[178] M. Baratchi, N. Meratnia, P. J. M. Havinga, A. K. Skidmore, and A. G. Toxopeus, "A hierarchical hidden semi-Markov model for modeling mobility data," in *proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (Ubicomp'14)*, Seattle, Washington, 2014, pp. 401-412.

A trajectory model gives us the possibility of taking the observations in form of mobility measurements and explain their meaning either in form of the path, the activity, or the means of transportation they are presenting.

Various spatio-temporal rules and dependencies are hidden in mobility data caused by different types of context variables such as the type and frequency of activities performed. A detailed model should encompass all these rules and dependencies. To better elaborate the spatio-temporal rules hidden in mobility data, an example from Dataset 3 is shown in Figure 4-1. Two visited grid cells, denoted by G_1 and G_2, have been extracted from a user's trajectory $\{o_t \mid t \in [1,T]\}$ in Geolife dataset [15-17]. Figure 4-1.a represents the probability $(P_h^{G_i})$ of user's presence (o_t) in these two different grid cells during different hours (h) of day $(P_h^{G_i} = \frac{|S_h^{G_i}|}{|S_h^+|}$ if $S_h^{G_i} = \{|o_t||o_t = G_i \& \, t \bmod 24 = h\}$ and $S_h^+ = \{|o_t||o_t = G_{j\in 1\ldots n} \, \& \, t \bmod 24 = h\}$. Figure 4-1.b represents the probability distribution of the duration of visits to these cells and Figure 4-1.c is the probability that visit to one place is followed by visit to the other in one hour.



	G1	G2
G1	-	0
G2	1	-

(c)

Figure 4-1 An example depicting spatio-temporal rules in a mobile object's history, (a) presence probability in two grid cells over 24 hours, (b) duration distribution of visits, (c) transition probability from one place to another

A number of rules can be extracted from these images. It can, for instance, be seen from the duration distribution graph, that presence of the user in each of these grid cells has a certain distribution (rule of duration of visits). Another interesting point is existence of a dependency in visits to these places. Although presence in each of these grid cells is of low probability, all visits to grid cell G_2 are followed by visit to grid cell G_1, while no visit to G_1 has been followed by a visit to G_2 (rule of transition between grid cells). This information can be used to efficiently model movement of this particular entity. Such model can be later applied for different purposes such as predicting future movements or to identify changing points in the mobility habits, or even to provide an adaptive sampling technique.

Designing a model, which can capture all the above-mentioned dependencies from real-life mobility datasets, is a challenging task. Firstly, trajectories are formed by components with different speeds (stay-points and transitions) being repeated with different frequencies. A model, which only captures frequency of visit to places, turns out to be biased to stay-points [179, 180]. On the other hand, preprocessing trajectories to take out segments with similar speed is time and energy consuming. Secondly, mobility data are extremely sparse and noisy. The sparseness, is sometimes caused by the system designer, as a tradeoff between accuracy and lifetime. Other times, it is caused by technical issues such as device mal-function. Mobility data is also noisy due to multipath and atmospheric effects.

The problem of trajectory modeling has so far been addressed through two perspectives, i.e., (i) determinist (through partitioning and clustering), and (ii) probabilistic (probabilistic modeling).

In the deterministic perspective, the approach towards modeling trajectories is to segment trajectories, assign segments to the clusters using a similarity measure, and calculate the frequency of usage of each cluster [164, 181-184]. Eventually, the observations are explained as the most probable cluster they belong to. Trajectory segmentation as mentioned above is simple and straightforward. However, it is time and energy consuming. Furthermore, such deterministic approach is useful when there exists a-priori knowledge about mobility patterns in the data (stay-points, duration of stay in stay-points, and etc.).

When not enough information over mobility data is available, probabilistic modeling technique can be used to discover patterns [179, 180, 185, 186]. These techniques generatively model trajectories as the sequence of observations that are representing a higher-level state.

4.1.1 Contributions

The general approach we follow in this chapter for modeling trajectories is answering the following questions: *What are the building blocks of a trajectory and*

how do mobility transitions happen? We address these questions from two viewpoints. The basis of our first approach is segmentation and clustering of trajectories and counting the frequency of occurrence of certain patterns, while the second approach utilizes probabilistic state-space modeling techniques to identify patterns. In other words, the first approach is more deterministic, whereas the second one is probabilistic.

With respect to deterministic trajectory modeling through partitioning and clustering, our contributions are as follows:

- We propose a two-leveled grid based clustering approach based on semantic and geographical data to find the frequently visited paths by mobile entities to the finest level of similarity.
- We use the concept of collective knowledge to deal with the uncertainty of trajectory representation when the level of uncertainty caused by missing samples and discrete sampling is increased.
- We evaluate our trajectory-clustering algorithm in comparison with Traclus Algorithm [181] (which is also based on trajectory partitioning) and show that our algorithm performs better in differentiating between trajectories in existence of uncertainties. We further perform a case study to see how this algorithm performs in finding frequent paths.

With respect to probabilistically modeling trajectories using state-space modeling techniques, our contributions are as follows:

- We propose a hierarchal hidden semi-Markov-based model (*HHSMM*) which can capture both frequent and rare mobility patterns in the movement of mobile objects.
- We apply the proposed model on real datasets and show how the model can find such patterns (e.g. frequent, rare, weekly) without a-priori knowledge about mobile object's behavior.
- We evaluate the performance of our model in terms of its correctness in prediction of mobility behavior and compare it with other spatio-temporal models.
- We test the sensitivity of the proposed model in presence of noise and missing measurements.

The remainder of this chapter is organized as follows. In Section 4.2, the related work in both of the above-mentioned approaches is presented. In Section 4.3 our deterministic model for finding frequent patterns in trajectories is discussed. The probabilistic state space modeling approach to find patterns in trajectories is presented in Section 4.4. Finally, in Section 4.5, we summarize this chapter.

4.2 Related work

4.2.1 Deterministic trajectory modeling

Previous research on deterministic trajectory modeling, mainly apply the commonly used data mining techniques such as clustering on mobility data. Trajectories have also been clustered using similarity measures [181, 182]. Some popular similarity measures which have been used to compare trajectories are Euclidian distance [187], LCSS (Least Common Subsequence) [188], DTW (Dynamic Time Warping) [183], ERP (Edit distance with Real Penalty) [189], EDR (Edit Distance on Real sequences) [182], and CATS (Clue Aware Trajectory Similarity) [190].

Recently, a few methods have also been proposed to deal with different notions of uncertainty in collected mobility data. In order to deal with the uncertainty caused by sampling error, a constant uncertain area around the trajectory points (cylindrical or square) is often considered [191, 192]. The problem of uncertainties in trajectories is also addressed by proposing a variant of fuzzy C-Means clustering algorithm [193]. Without getting help from a complementary mechanism such as a sliding window, the previously mentioned similarity measures can only find the similarity between complete trajectories but not between common sub-trajectories. In order to build a deterministic model, knowledge about similar sub-trajectories is needed.

To be able to find similar fractions of trajectories, Traclus [181] was proposed to find the common portions of trajectories (sub-trajectories) by first partitioning them based on the movement behavior and then clustering these trajectory partitions. Another approach to finding frequent trajectories is by considering the semantic information such as stay-points [164, 184, 194] and finding frequent patterns in semantically defined trajectories. Stay-points are spatial neighborhoods were the mobile entity has a speed near zero. However, when semantic information is used for segmenting trajectories, trajectory-clustering techniques do not perform as robust when uncertainties such as missing samples and noise in the trajectories increase (this problem is shown with an example in Section 4.3). In our deterministic approach, we use collective knowledge of trajectories to reconstruct trajectories and cluster them to the finest level of similarity.

4.2.2 Probabilistic trajectory modeling

Recently, a number of probabilistic models have been proposed to model the movement of mobile objects. These models attempt to capture the variation in spatial dependencies. An ensemble method has been used by [186] to probabilistically model the movement on frequently visited places considering

different context variables. Authors of [195, 196] have used topic models to learn mobility patterns from long duration sequences. A probabilistic kernel method is proposed in [197] to predict future locations. Different versions of Markov models have also been applied on mobility data. For instance, Order-k Markov model was used in [198] to predict the movement of users in Wi-Fi network cells. In [180], a model based on hidden Markov models is proposed for modeling movements from one stay-point to another, while authors of [199] used mixed Markov model for the same purpose. A mixed autoregressive hidden Markov model is proposed in [179] on stay-points.

The main drawback of these methods is that they do not completely consider the temporal variability in the mobility data. In these models, a trajectory is only partially modeled either as a sequence of visited stay-points or just as the transition path between stay-points. Apart from being incomplete, these methods require pre-processing the data to extract regions of interest or stay-points. The required pre-processing phase is rather time/energy inefficient. The above-mentioned problem is caused due to the inherent limitation of hidden Markov model, which is considering constant duration for each system state. Therefore, there is still need for a model, which can be applied on complete mobility data, consisting of both stay-points, and transitions by considering their temporal variability.

Hidden semi-Markov model addresses the above-mentioned issue by considering an additional duration property for each state. To the best of our knowledge there is only one previous research [185] which has considered using hidden semi-Markov model on mobility data. However, the authors of this research have only evaluated their model on a synthetic dataset representing data of few hours. As we will show, when modeling large dataset of human mobility, composed of complex patterns (e.g. weekly), the technique used in [185] results in a very course grained model.

4.3 A two-leveled deterministic trajectory model

Regions of interest and stay-points can provide a good basis for segmenting trajectories. After trajectories are broken into smaller sub-trajectories based on stay-points a clustering algorithm can be used to find the frequencies. While considerable attention has so far been paid to find similar trajectories considering the entire trajectory [181, 182], not much attention has been paid on finding similar sub-trajectories between regions of interest (e.g. between semantic areas).

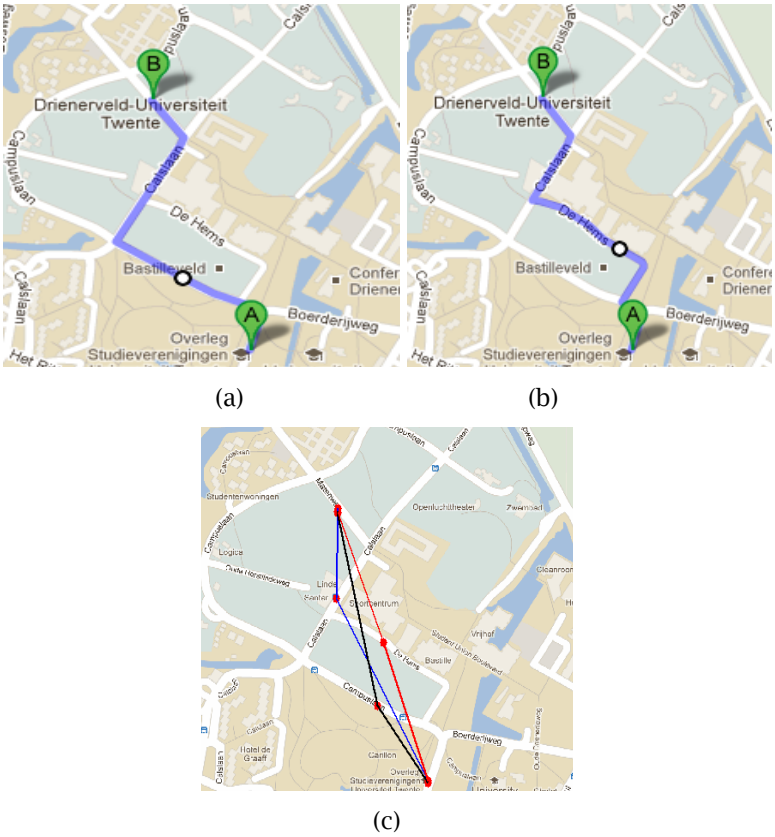


Figure 4-2 (a) Path 1 (b) Path 2 (c) trajectories representing Path 1 and Path 2

The semantically similar sub-trajectories may represent different (physical) paths. When trajectories are partitioned and grouped based on stay-points, the sub-trajectories in each partition are already similar with respect to their origin and destination and potentially have common smaller sections (referred to as sub-paths). Therefore, identifying these different sub-paths in a group of trajectories, which are to some extent similar in the path they present, is difficult. Additionally, different sources of uncertainty in trajectories, such as noise and missing samples, make this procedure even more challenging. In Figure 4-2, the challenge in distinguishing between trajectories, which represent different paths, is shown. Figure 4-2.a-b represents two different paths between two spatial spots (A and B). Figure 4-2.c shows three different trajectories between these spatial points resulted by interpolating the measured spatial points. Black and red trajectories each follow one of these paths and the blue trajectory is not classifiable due to having a crucial sample missing.

Other than the challenge mentioned above, there exists another challenge in representing trajectories from mobility samples in presence of uncertainties such as missing samples and noise. One of the most popular methods in presenting uncertain trajectories is interpolating two consecutive GPS measurements. Depending on the sampling rate, number of missing samples, and their position, interpolation of GPS measurements of two similar trajectories may result in two completely different trajectory representations. Trajectory representation while dealing with the uncertainties can also be performed through correcting trajectories using the collective knowledge [200]. The collective knowledge in this case is the knowledge gained by considering mobility data of all trajectories on a specific path. This knowledge is achieved by aggregating all the points on different trajectories. As seen in Figure 4-3, the red points can better represent the frequently visited paths than the blue interpolating lines. The knowledge extracted by aggregating all red dots, is the collective knowledge.

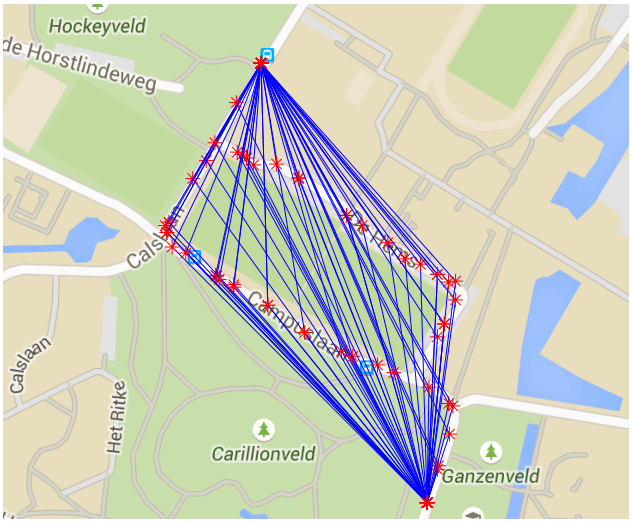


Figure 4-3 Individual trajectories (blue lines) alone do not provide enough information to construct the route while their aggregation (red dots) can help us reconstruct the route more accurately

In order to build a trajectory model, we are interested in segmenting trajectories and counting their frequency of repetition. These segments represent the building blocks of trajectories and their differences. In terms of finding partitions which are shared by different trajectories, our work is more in-line with Traclus [181]. While Traclus does not take the uncertainty of trajectory representation into account, we deal with this notion of uncertainty through using the collective knowledge of trajectories.

4.3.1 Problem definition

Let us assume a trajectory database denoted by $D = \{Tr^1, \dots, Tr^m\}$, composed of trajectories $Tr^i = P^i_{t_1} \dots P^i_{t_{n_i}}$, where i is the trajectory ID and the indexes $(t_1 \dots t_{n_i})$ represent the time-stamps of samples representing a sequence of spatial points visited by a mobile entity during a day. We aim to find frequently traversed paths and sub-paths between stay-points. The length of trajectories is variable and there are missing samples due to different reasons (hardware failure, environmental conditions, etc.).

We use the following definitions:

Definition 4.1: *Sub-trajectory* $STr^i = P^i_{t_1} \dots P^i_{t_{n_i}}$ *(the mobility data between two semantic places) is a fraction of a trajectory which has its first point in one stay-point (origin) and its last point is in another (destination).*

Definition 4.2: *Path* $PA^i = \{SP^1 \dots SP^k\}$ *is a group of sub-paths (retrieved from the collective knowledge) that represent a real-world equivalent of route from an origin to a destination (for instance a street segment).*

Definition 4.3: *Sub-path* $SP^i = \{g_i \dots g_n\}$ *is a section of a path composed of a list of cells* (g_i) *on a grid. Sub-paths can be considered as units of difference between paths.*

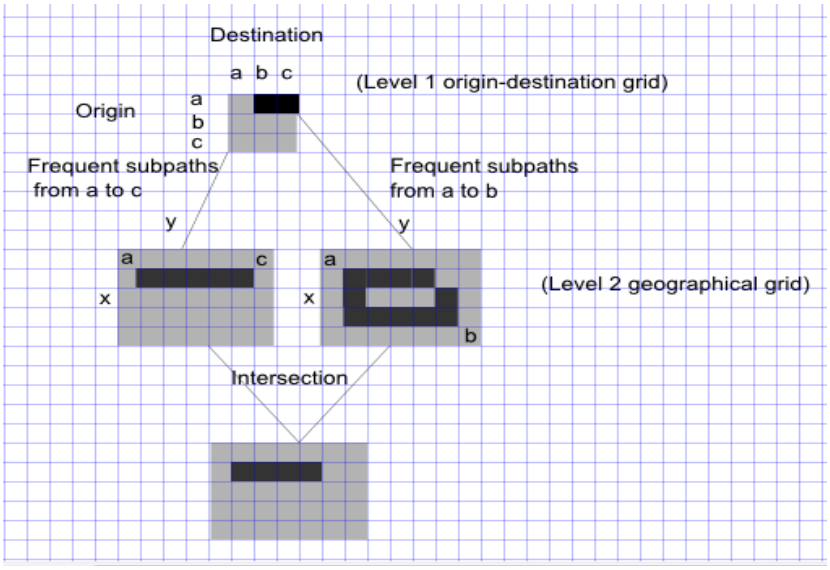


Figure 4-4 Overview of our two-level approach

4.3.2 Methodology

Grid based clustering methods quantize the object space into a finite number of grid cells on which all of the operations for clustering are performed [201]. We use two levels of such grids as shown in Figure 4-4, to find frequently visited paths and sub-paths. Our technique consists of three steps, (i) sematic-based clustering, (ii) geographical-based clustering, and (iii) cluster intersection.

4.3.2.1 Level 1: semantic-based clustering

We use the semantic information acquired from a semantic grid which groups trajectories based their origin and destination. This way, we first cluster the sub-trajectories based on the information we can achieve from the stay-points in which a person stays longer than a predefined threshold. Such stay-points are extracted by the method proposed in [202], in which each stay-point is a place where the speed of the person is near zero. Then we group sub-trajectories into clusters such that each cluster contains sub-trajectories between the same set of origin and destination. A grid cell in the semantic grid can store, for instance, all the sub-trajectories traversing from home (origin) to work place (destination). (Existence of a pattern mining layer on top of the semantic grid is also possible to find the frequent semantic trajectories [192]).

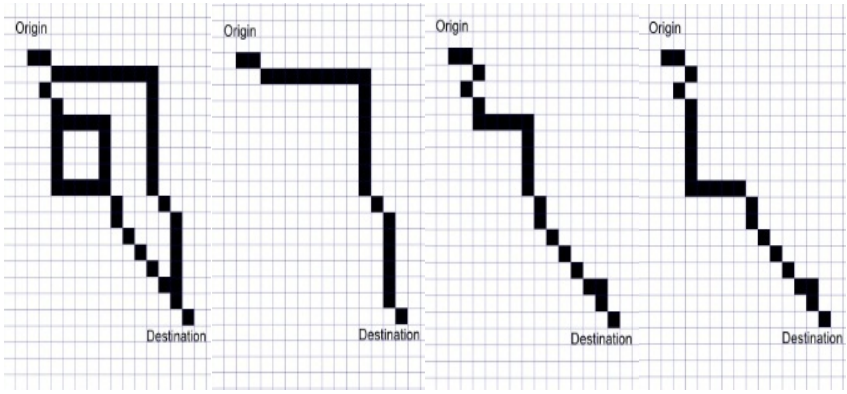
4.3.2.2 Level 2: geographical grid-based clustering

In the next level, we cluster sub-trajectories to find frequently visited paths and sub-paths between each pair of origin and destination. The challenge to face here is that these sub-trajectories are already somewhat similar (as they have the same origin and destination). Therefore, it is necessary to first find the source of difference between them. Additionally, some sub-trajectories have missing points which make their correct representation difficult and consequently make them unclassifiable with respect to different paths traversed.

To address these challenges we follow four steps. First, we aggregate all sub-trajectory points (from the same semantic grid cell) to find a connected neighborhood between the origin and destination based on the common knowledge of sub-trajectories. Next, we find the source of difference between paths (sub-paths) in such neighborhood. Later, we find the order of subpaths in a path. Eventually, we redefine and group sub-trajectories with respect to these units of difference. These procedures are better explained in the following sections.

Step 1: Finding connected neighborhoods: We form a geographical grid of size $M \times M$. Having the start time of all sub-trajectories synchronized, each grid cell denoted by g_i, $(1 < i < M^2)$ will hold the number of sub-trajectories which have a

point on it, denoted by $c(g_i)$, along with the median of their time index, denoted by $m(g_i)$. With this median we can later keep the order of sub-paths.



(a) (b) (c) (d)

Figure 4-5 (a) A connected neighborhood between a pair of origin and destination and, (b-d) 3 frequently visited paths, inferred from collective knowledge achived by aggregating all trajectories

Assuming that there is no backwards movement between the origin and destination, if we only filter the cells $\{g_i|\ c(g_i) >$ defined threshold$\}$, the layout of frequently visited paths between the origin and destination will become visible (in form of a connected neighborhood) based on the common knowledge of sub-trajectories. As seen in Figure 4-5, the paths that form a connected neighborhood may have some sub-paths in common and some different sub-paths. These sub-paths can be considered as the unit of (dis)similarity between two paths. Therefore, we need to break down the connected neighborhood into sub-paths to be able to use them for defining the frequent paths.

Step 2: Finding sub-paths in connected neighborhoods: A frequently visited path can be represented as an ordered list of sub-paths. An idea to find sub-paths in the connected neighborhood is to find breakpoints where a group of paths meet each other (converge) or where they separate from each other (diverge). Afterwards, the connected neighborhood between these breakpoints can be defined as sub-paths.

Algorithm 4.1 (SPdefine) explains how we find sub-paths between the origin and destination. This algorithm starts traversing the grid from the cell, which represents the origin. It follows the path on the grid by iteratively selecting neighbors (referred to as selected cell(s) (SC)) and moving forward following the adjacent neighbors.

Algorithm 4.1 (SPdefine)

INPUT: S (start point), TH (threshold), $G = \{g|c(g) > 0$ (grid cells)$\}$
OUTPUT: $LSubPaths$ (List of subpaths)

```
1:  Queue ← ∅, LBP ← ∅; //LBP is a list of breakpoints
2:      Add S to LBP & add {N| N is non-visited non-adjacent neighbor of S &
3:      c(N) > TH} to Queue ;
4:  While Queue is not empty
5:      Do SC ← Dequeue a cell from Queue;
6:      While SC is visited;
7:      Visit SC;//setting a visit flag for each cell
8:      Add SC to TempSubPath;
9:      NonAdjacentNeighbors ← {N| N is non-visited non-adjacent neighbor of SC
10:     & c(N) > TH};
11:     Intersection ← False ;
12:     While length (NonAdjacentNeighbors ) == 1 & ! Intersection
13:         Visit NonAdjacentNeighbors(1);
14:         Add NonAdjacentNeighbors(1) to TempSubPath;
15:         SC ← NonAdjacentNeighbors(1);
16:         NonAdjacentNeighbors ← {N| N is non-visited non-adjacent neighbor of
17:         SC & c(N) > TH};
18:         VisitedNeighbors ← {N| N is visited neighbor of SC};
19:         If any of VisitedNeighbors are in (LBP)
20:             Intersection ← True;
21:         End if
22:     End while
23:     If | NonAdjacentNeighbors | > 1
24:         Add NonAdjacentNeighbors(1..n) to Queue;
25:         Add SC to the LBP;
26:     End if
27:     Add TempSubPath to the list of LSubPaths
28: End While
```

The Adjacent/non-adjacent neighbor concept is depicted in Figure 4-6. The reason for choosing this concept is to be able to select more than one grid cell to move forward to whenever necessary. This happens, for instance, when the width of a path is more than the width of a grid cell. Adjacent neighbors are neighbors of selected cell(s), which have a common edge, and non-adjacent neighbors are those neighbors without a common edge. A group of adjacent neighbors should be

considered as one non-adjacent neighbor for the selected cell. Therefore, when we choose neighbors as the next round's selected cell(s), a group of adjacent neighbors might be chosen.

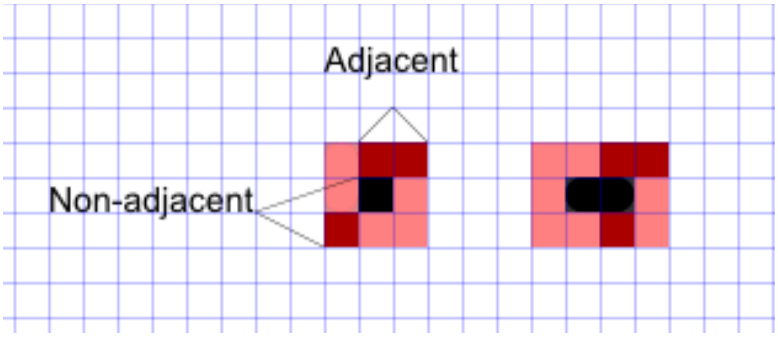


Figure 4-6 Left: a selected cell in black with three neighbours (dark red), two of them are adjacent with each other, one is non-adjacent with the other two. Right: two selected cells (black) and their two non-adjacent neighbours (dark red)

In each iteration, the algorithm extracts a selected cell(s) from the queue, adds it to the start of a sub-path and checks its neighbors. While there is only one unvisited non-adjacent neighbor (N) (with $c(N) > TH$) and no breakpoints in the neighbors, selected cell(s) will be added to the sub-path (lines 12-22). In case there are more than one adjacent cell to move forward to, the selected cell(s) will be added to the list of break points, the non-adjacent neighbors will be added to the queue (the order of cells is not important), the current sub-path will be terminated and added to the list of sub-paths (lines 23-27).

After a while, some sub-paths may have cells from both of their ends in the queue (a cell from their start and a cell from their end). In order to avoid traversing the sub-paths twice, when removing cells from the queue we will only select the cell(s), which are not already visited (lines 3-5). Finally, we will have a list of sub-paths in which each sub-path is defined by a list of cells denoted by $\{g_i | \, i \in M \times M \}$ and one or two breakpoints (one on each end).

Step 3: Ordering sub-paths in the tree of sub-paths: After finding sub-paths in a connected neighborhood between a pair of source and destination, we order them using a tree structure (tree of sub-paths). Matching the breakpoint in the beginning and end of each sub-path performs this ordering. As seen from Figure 4-7, the ordered sequence of sub-paths from the route to the leaf of the tree shows the frequent paths from origin to destination.

In this figure, the frequent paths inferred from the tree of sub-paths are: PA^1={SP^1, SP^2, SP^7}, PA^2={$SP^1, SP^3, SP^5, SP^6, SP^7$}, PA^3={$SP^1, SP^3, SP^4, SP^6, SP^7$}.

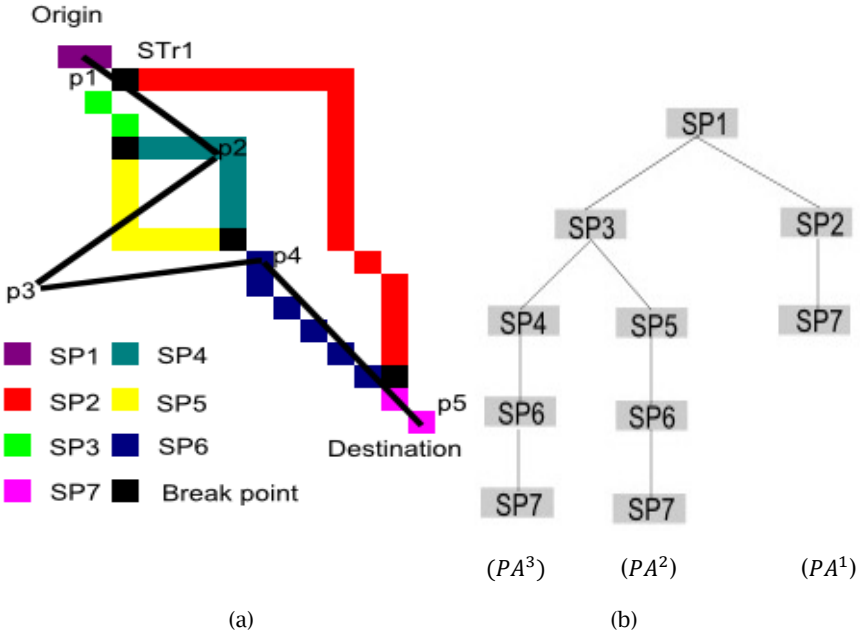


Figure 4-7 (a) A connected neighborhood between a pair of origin and destination with 7 sub-paths, (b) the representative tree of sub-paths

Step 4: Redefining trajectories based on sub-paths and clustering them based on the tree of sub-paths: After fragmenting a neighborhood into the frequently visited sub-paths and finding the frequently visited paths using the tree of sub-paths, we redefine each sub-trajectory in terms of sub-paths. We read the points on each sub-trajectory in their temporal order. If a point was on or near (in the neighboring cells) a sub-path, it will be replaced by that sub-path. This means that, existence of only one point in the sub-path indicates that this sub-trajectory passes through the sub-path completely. Other points that are not on any sub-path will remain intact (in case we are interested in outliers too). For instance, looking at Figure 4-7, the new representation of sub-trajectory STr^1 will be {SP^1, SP^4, P^3, SP^6, SP^7}.

If we consider that each path in the tree is the core of a cluster, then the final step is to assign the redefined sub-trajectories to the correct cluster. For the comparison we have chosen to use a similarity measure similar to *LCSS* [188], which ranks the matching parts between two time series based on their similarity.

If $SP_{1\dots m}$, and $SRT_{1\dots n}$, denote the list of sub-paths from a path (cluster), and the sub-path list (only sub-paths and not points in case of STr^1, the sub-path list will be $\{SP^1, SP^4, SP^6, SP^7\}$ of a redefined sub-trajectory, then the similarity measure between the redefined sub-trajectory and a path (cluster) formed by the tree is:

$$SM = \begin{cases} 0 & if\ m = 0\ or\ n = 0 \\ SM\big(rest(SP), rest(SRT)\big) + 1 & if\ (SP_m = SRT_n) \\ SM(rest(SP), SRT) & otherwise \end{cases} \quad (4.1)$$

This way, each sub-trajectory is compared with all clusters and assigned to one or a number of clusters based on the maximum similarity. The points that remain on the redefined sub-trajectory can be ignored (considered as noise).

During this procedure we can also score the sub-paths. The score of a sub-path SP_i denoted by $Score(SP_i)$ is the number of the sub-trajectories that have followed it. If the maximum similarity measure of a sub-trajectory and the paths (clusters) is owned by one path (cluster), then the sub-trajectory belongs to that path (cluster) and the score of all sub-paths on the path are incremented by one. If the maximum similarity measure of the sub-trajectory is shared by a number of paths (clusters), that sub-trajectory is uncertain between those paths (clusters). We can increment the score of these sub-paths by 1/|paths with maximum similarity measure|. Sub-trajectories that have equal similarity measure to all paths (by only following the sub-paths on the start and end) and those with a considerable number of remaining points are outliers.

Let us consider the tree shown in Figure 4-7. A sub-trajectory, which is redefined as $\{SP^1, SP^5, SP^7\}$ has a similarity measure of 2 to PA^1 and PA^3 (shown in the figure) and similarity measure of 3 to PA^2. Therefore, it will be clustered with PA^2. This increases the score of each sub-path on PA^2 by one. A sub-trajectory represented by $\{SP^1, SP^3, SP^3, SP^7\}$ is similar to PA^2 and PA^3 with a similarity score of 3 while its similarity score to PA^1 is 2. Therefore, it will be considered as uncertain between PA^2 and PA^3, and cause the score of each sub-path on these paths to increment by ½ score.

4.3.2.3 Level 3: cluster intersection

After all the sub-paths are scored with respect to the sub-trajectories that follow them, for each cell of the semantic grid (each pair of source and destination), we compare the sub-paths from one cell of the semantic grid to the sub-paths of other cells. In case the intersection of two sub-paths (in terms of the id of geographical grid cells) is not empty, the intersected sub-path $SP_c = \{g_i | g_i \in SP_a \cap SP_b\}$ will be added to the list of sub-paths with a score equal to the score of two sub-paths

($Score(SP_c) = Score(SP_a) + Score(SP_b)$). By so doing, we will have a list of scored sub-paths, out of which the top sub-paths can be chosen as the most frequently visited sub-paths.

4.3.3 Evaluation

4.3.3.1 Complexity analysis

In this section, we evaluate the complexity of Algorithm 4.1 (SPdefine). If we consider having N number of trajectories and P clusters (frequent paths), the complexity of the algorithm will be $O(NP)$. The complexity of Traclus [181], to which our approach is more similar, is $O(N^2)$. Therefore, when the number of frequent paths (P) shared by a large number of trajectories is limited, our approach performs more efficiently. The memory required for SPdefine to perform is dependent on the size of queue it uses. This makes its requirement equivalent to $O(G)$ (maximum queue size) with G representing the grid size.

4.3.3.2 Performance evaluation

In this section, we present performance evaluation of our model on data of a mobile entity from Dataset 1. As the first step, we extracted the stay-points where the person has stayed longer than 30 minutes. As shown in Figure 4-8, we then chose the group of sub-trajectories between two different staying points. The only previous clustering approach which addresses the problem of finding common sub-trajectories is Traclus [181]. Traclus first partitions trajectories based on the change in the behavior of trajectory (e.g. direction). It then clusters the resulted line segments using a line-based similarity measure. The behavior of sub-trajectories that we formed between two semantic places is quite similar. Therefore, we simply consider the sub-trajectory partitions as being the line segments achieved through interpolating consecutive measurements.

We compare 74 sub-trajectories between two semantic places shown in Figure 4-8-a. These sub-trajectories represent the two paths between two stay-points shown in Figure 4-8-b-c. It can be seen that some sub-trajectories have enough number of points to be assigned to Path 1 or Path 2, while some other have only points on the intersected sub-paths and cannot be clustered by human eye (see the example shown in Figure 4-2). Our goal is to distinguish between these two paths as two clusters, to find the number of sub-trajectories that have followed them and also to find their intersection as the most frequently visited sub-path.

As shown in Figure 4-8.d, Traclus will only find one cluster. The reason is that due to closeness of two different paths, there is small spatial difference between sub-

trajectories of two different paths and the uncertain trajectories fill this gap. In addition, existence of missing points and the spatial result in similarity between sub-trajectories. The tree of sub-paths formed by our method, however, can find two frequent paths (clusters) between the source and destination with 4 sub-paths. Traclus represents the trajectories by getting an average of the cluster by a sweeping mechanism. We, however, represented the sub-paths by getting the average of points in each grid cell and ordering them by their median of timestamps. Using this mechanism the representation of sub-paths is closer to their realistic representation. Moreover, our method can make a distinction between spatially close paths and uncertain sub-trajectories.

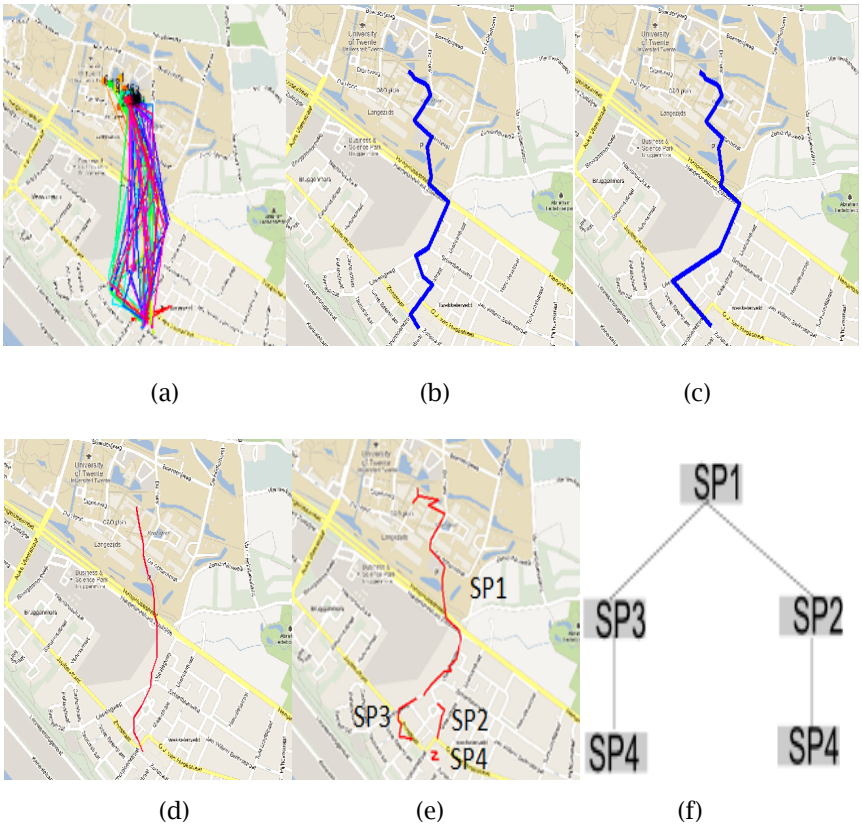


Figure 4-8 (a) group of sub-trajectories, (b,c) two different paths common in one subpath, (b) Path1, (c) Path2, (d) representation of the only cluster found by Traclus, (e) representation of 4 sub-paths identified by our method, (f) the tree of sub-paths

We performed experiments with 3 different grid sizes. Table 4-1 and Table 4-2 respectively represent the number of frequently visited paths, and their score of sub-paths found.

Clusters	Number of sub-trajectories assigned to the cluster			Number of sub trajectories in clusters
	GS= 70×70	GS= 80×80	GS= 100×100	
Path 1	13	22	22	24
Path 2	12	35	37	38
Uncertain	49	17	15	12

Table 4-1 The number of clustered trajectories being identified using the tree of sub-paths

Sub-path	Score of sub-path			Number of sub trajectories with a point in the sub-path
	GS= 70×70	GS= 80×80	GS= 100×100	
Sub-path 1	74	74	74	74
Sub-path 2	40.5	30.5	29.5	24
Sub-path 3	37.5	43.5	44.5	38
Sub-path 4	70	74	74	74

Table 4-2 The score of sub-paths

In each table, the last column shows the actual values, which were measured by analyzing sub-trajectories visually. Table 4-1, shows that the precision of the method increases as the grid size decreases. This is due to the fact that, the smaller the cell is, the easier it is to precisely represent the start and end of a sub-path. Therefore, the sub-trajectories with points only near the start and end of a sub-path are better assigned to the right sub-path. It is inferred from Table 4-2 that the scores of Sub-paths 2 and 3 are higher than the actual number of sub-trajectories that have a point on them. The reason is that, there are 12 unclassifiable trajectories between the two paths, which we chose to split their score between their sub-paths. Also, it is seen that with grid size 70×70 the number of sub-trajectories in Sub-path 4 is less than the total number of trajectories. This is due to the fact that, with this grid size, some of the sub-trajectories do not have a point in this small sub-path. This is a rare case where a stay-point covers a wider area than the sub-path.

4.3.3.3 Different approaches and their desirable properties

Table 4-3 shows a comparison between different clustering approaches in terms of their support for different desirable properties in finding frequent sub-paths between semantic places.

Generally, methods that are interpolation-based are sensitive to uncertainties caused by discrete sampling and missing points. Methods that do not rely on a framework for partitioning trajectories are not able to find their similar sections in trajectories. Noisy measurements can be ignored by a non-metric system of comparison (not measuring the distance of points but counting the number of similar points) [187] or density based clustering. In our two-leveled approach, we ignore noisy measurements by scoring the similarity between trajectories and frequent paths. We use collective knowledge of sub-trajectories to redefine them. Doing so, we deal with trajectory representation uncertainties caused by discrete sampling and the problem of missing samples. We also address the measurement errors to some extent by defining adjacent neighbors to move forward to and assigning points on sub-paths, if they are close to them.

Method	Sub-trajectory based clustering	Noise	Missing samples & Discrete sampling	Measurement error
EDR [182]	No	Yes	No[6]	No
Traclus [181]	Yes	Yes	No	No
[193]	No	No	No	Yes
SPdefine	Yes	Yes	Yes	No

Table 4-3 Comparison of different trajectory clustering approaches with respect to different desirable properties

4.3.3.4 Case study using Dataset 2

In this section, we perform experiments with the Capricorn dataset. Data from three Capricorns are available. In order to perform these experiments, we extracted stay-points using the methods explained in Section 4.3.3.2. Apart from the accidental sparseness, caused by technical issues; in this dataset selected samples are missing using a predefined duty cycle. Due to this sparsity, we looked for stays of over 5 hours in neighborhood of 200 meters.

As seen from Figure 4-9, using the selected parameters, there are only three different stay-points for Entity 1. For the other Capricorns, only one distinguishable stay-point is found. Using Algorithm 4-1 and a grid with the size of 10×10, we found the most frequent path used between the two stay-points (Shown in Figure

[6] EDR has a mechanism to deal with gaps in data, but in case the paths are close to each other and have common sub-paths it will not be able to deal with the uncertainty of missing samples and discrete sampling

4-9 by red and blue) for Entity 1. The results of this experiment are presented in Figure 4-10.

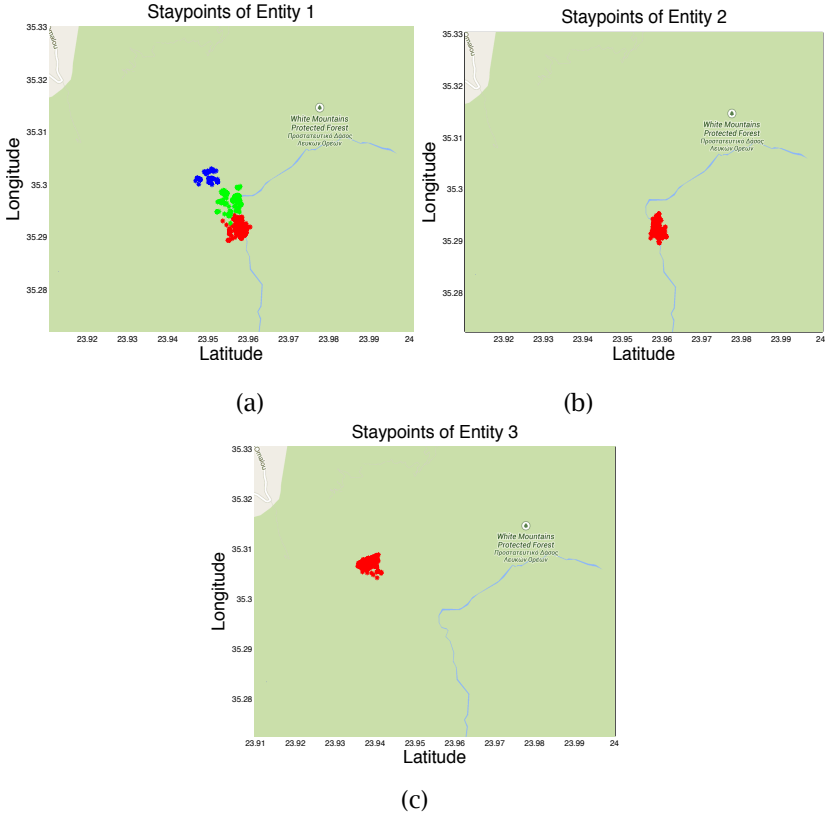


Figure 4-9 Staypints extracted from the entities in the Capricorn dataset

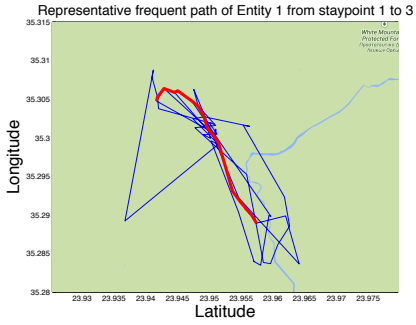


Figure 4-10 The most frequent path used by Entity 1 in the Capricorn dataset

4.4 A hierarchal probabilistic trajectory model

In Section 4.3, we proposed a trajectory-clustering approach, which could breakdown complex trajectories using semantic information from stay-points. Although this approach captures short-term dependencies in trajectories, it is yet partial, as it does not consider other factors that explain the dynamics of a trajectory. Some of these factors are the duration of stays, speed, or longer-term dependencies. Furthermore, the previous approach relies on prior information form stay-points such as minimum stay-duration.

Considering what mentioned above, in order to make such form of deterministic models complete, many assumptions need to be made by the person who writes/runs the algorithm.

To address the drawbacks of the previous approach, we further studied other modeling approaches, which do not strictly rely on assumptions. In the rest of this chapter, we use a probabilistic modeling approach and propose a hierarchical hidden semi-Markov model offering the following advantages: (i) it efficiently captures temporal characteristic of visits to places, (ii) its hierarchical structure enables it to find complex patterns, (iii) its granularity is adjustable to resources available, and (iv) it requires minimum assumptions made by the user.

4.4.1 Problem definition

Let us assume that the movement dataset $O = \{o_{t_i} \dots o_{t_n}\}$ of a mobile entity over several days is given. This dataset is composed of chronologically ordered two-dimensional geo-spatial points representing object's location at time-stamp i, where the distance between t_i and t_{i+1} is variable. We are looking for a model λ composed of a number of spatio-temporal rules relating the noisy and unevenly sampled movement data to their context related state $Q = \{q_{t_i} \dots q_{t_n}\}$. These states can represent the activity that governs the movement.

4.4.2 Background

We begin this section by providing background information on different state space models and their parameters.

4.4.2.1 State space models

Hidden Markov model is a dynamic Bayesian network highly capable of representing data in form of temporal sequences of discrete states. Observations in this model are described in terms of a number of unobservable states with higher-

level semantic concepts, in which transition from one state to another happens on the basis of a specific probability distribution [203].

The hidden Markov model λ composed of M number of states $\{s_1 \dots s_M\}$ and N number of unique observations $\{v_1 \dots v_N\}$ is defined by a number of parameters (Eq. (4.2)):

$$\lambda = (Q, O, A, B, \pi) \tag{4.2}$$

In the above model, $O = \{o_t | t \in T\}$ and $Q = \{q_t | t \in T\}$ represent the entire observation sequence, and the entire high-level state sequence, respectively (T is the set of uniformly distanced timestamps). A is the $M \times M$ state transition probability matrix representing the probability of transition between states expressed as ($a_{ij} = P[q_{t+1} = s_j | q_t = s_i]$). B is $M \times N$ emission probability matrix representing the conditional probability between states and observations ($b_i(v_j) = P[o_t = v_j | q_t = s_i]$), and π is the initial probability distribution vector of size $M \times 1$, ($\pi = P[q_1 = s_i]$).

In the original hidden Markov model, due to the first order Markov assumption, it is implicitly assumed that the duration of each system state is constant or exponentially distributed. As a consequence, in these models transition between states happens at any time and self-transition is allowed.

Apart from simplicity and flexibility offered by the model, its downside is that it does not take any advantage of the information hidden in the duration of visit to different places. Stay-points and transition paths have different duration distributions, which are also needed to be taken into account. In order to deal with this problem, later the original hidden Markov model was extended to hidden semi-Markov model, where apart from the transition between states there is an additional parameter for explicitly modeling the duration of states.

The hidden semi-Markov model (also known as explicit duration hidden Markov model or variable duration hidden Markov model) is represented by $\lambda = (Q, O, A, B, C, \pi)$. In this extended model, C is the additional important ($M \times D$) matrix added to the previously mentioned parameters of hidden Markov model where D is the maximum state duration and $c_i(d) = P(c_{s_i} = d)$ represents the probability of state s_i last for d time units. This type of model is previously used for presenting a sequence of events with different duration for instance, in video image processing, and daily activity modeling.

Given an output sequence in form of a sequence of observations; a parameter-learning algorithm is performed to estimate the parameters of the model λ. The best set of state transitions, output probabilities, and state duration matrices is estimated in this way.

4.4.2.2 Model parameter estimation

Estimation of the transition and emission probabilities in a hidden Markov model is performed by iterative re-estimation of the model parameters until a maximum likelihood is achieved. One of the well-known decoding algorithms used for this purpose is Baum-Welch algorithm [203]. In each iteration of this algorithm, forward $\alpha_t(m,d)$ and backward variables $\beta_t(m,d)$ for each state s_m at time t with duration d are calculated and the new parameters are re-estimated by maximizing the likelihood of the posterior probability density over the model parameters. Among different variations of Baum-Welch used for modeling with hidden semi-Markov model, we have chosen the method proposed in [185] as it considers missing observations.

Assuming that τ_t denotes the remaining time of the current state q_t, then the forward variable $\alpha_t(m,d)$ which is the probability of the system being at state s_m, with remaining time d at time t is calculated by (Eq. (4.3)):

$$\alpha_t(m,d) = Pr\left[o_1^t, (q_t,\tau_t) = (s_m,d)\right] \tag{4.3}$$

Achieved by the recursion formula in Eq. (4.4.). As seen in this equation, in case the observation is not valid, by using $\alpha_{t-2}(m,d+1)$ implicitly equal probability is considered for all states:

$$\alpha_t(m,d) = \begin{cases} \alpha_{t-1}(m,d+1)b_m(o_t) & t \in T \\ \alpha_{t-2}(m,d+1) & t \notin T \end{cases} \tag{4.4}$$

Where the initial condition is measured using Eq. (4.5):

$$\alpha_1(m,d) = \pi_m b_m(o_1) p_m(d) \tag{4.5}$$

The backward variable is defined as Eq. (4.6):

$$\beta_t(m,d) = Pr\left[o_{t+1}^T | (q_t,\tau_t) = (s_m,d)\right] \tag{4.6}$$

With the recursion formula (Eq. (4.7), Eq. (4.8)) and initial condition (Eq. (4.9)) defined as:

$$\beta_t(m,d) = \begin{cases} b_m(o_{t+1})\beta_{t+1}(m,d-1) & t \in T \\ \beta_{t+1}(m,d-1) & t \notin T \end{cases} \tag{4.7}$$

$$\beta_t(m,1) = \sum a_{mn} b_n(o_{t+1}) \left(\sum_{d>1} p_n(d)\beta_{t+1}(n,d) \right) \tag{4.8}$$

$$\beta_T(m,d) = 1, (d \geq 1) \tag{4.9}$$

The other variables defined below are used later in estimation of model parameters:

$$\xi_t(m,n) = Pr[o_1^T, q_{t-1} = s_m, q_t = s_n] = \alpha_{t-1}(m,1)a_{mn}b_n(o_t).\sum_{d\geq 1} p_n(d)\beta_t(n,d) \tag{4.10}$$

$$\gamma_t(m) = Pr[o_1^T, q_t = s_m] = \gamma_{t+1}(m) + \sum_{n\neq m}\left(\xi_{t+1}(m,n) - \xi_{t+1}(n,m)\right) \tag{4.11}$$

$$\begin{aligned}\eta_t(m,d) &= Pr\,[o_1^T, q_{t-1} \neq s_m, q_t = s_m, \tau_t = d] \\ &= \left(\sum\nolimits_{n\neq m} \alpha_{t-1}(n,1)a_{mn}\right) b_m(o_t)p_m(d)\beta_t(m,d)\end{aligned} \tag{4.12}$$

By maximizing the a-posteriori probability path, parameters of the model are inferred. Estimation and re-estimation of model's parameters can be performed through the following equations [204]:

The maximum a posteriori estimate of state q_t (Eq. (4.13)) is:

$$q_t = arg\ \max_{1\leq m\leq M} Pr[q_t = s_m|\, o_1^T] = arg\ \max_{1\leq m\leq M}\ \gamma_t(m) \tag{4.13}$$

The maximum likelihood re-estimate of initial state π_1 is:

$$\pi_1 = \gamma_1(m)/G_1 \tag{4.14}$$

With G_1 as a normalization constant ($\sum_1^m \gamma_i(m)$). The maximum likelihood re-estimates of the transition probability a_{mn} ($n \neq m$) is calculated by Eq. (4.15):

$$\hat{a}_{mn} = \sum_1^T \xi_t(m,n)\,/G(m) \tag{4.15}$$

The maximum likelihood re-estimate of the state duration is (Eq. (4.16)):

$$\hat{c}_m(d) = \sum_1^T \eta_t(m,d)\,/H(m) \tag{4.16}$$

Where $H(m)$ is a normalizing constant calculated as (Eq. (4.17)):

$$H(m) = \sum_{d=1}^{D}\sum_{t=1}^{T} \eta_t(m,d) \tag{4.17}$$

The re-estimation of observation of v_k over given state for $o_t = v_k$ is:

$$\hat{b}_m(v_k) = \sum_1^T \gamma_t(m).\,\delta(o_t - v_k)/V(m) \tag{4.18}$$

With $V(m)$ as a normalization constant calculated by (Eq. (4.19)):

$$V(m) = \sum\nolimits_k \sum_1^T \gamma_t(m)\delta(o_t - v_k) \tag{4.19}$$

For $o_t = v_k$, $\delta(o_t - v_k)$is equal 1 and it is 0 otherwise.

The above-mentioned equations are used in learning the parameters of the Hidden semi-Markov model. The iterative procedure of parameter learning is explained algorithmically in Algorithm 4.2 (TrainHSMM) based on [204].

Algorithm 4.2 (TrainHSMM)

INPUT: M (Maximum number of States), D (Maximum state duration), $O_{1\ldots T}$ (Observation sequence), $maxIter$ (Maximum number of iterations)
OUTPUT: A (State transition matrix), B (Emission matrix), C (State duration distribution matrix), π (initial state probability matrices), Q (State sequence)

For i=1 **to** $maxIter$
    Calculate the initial forward variable;// Eq. (4.5)
    For t=2 **to** T
        Calculate the forward variable; // Eq. (4.4)
    End for
    If the model reached a desired level of convergence **break**
    For t=T-1**to** 1
        Calculate ξ_t, γ_t, η_t;// Eq. (4.10)(4.11)(4.12)
        q_t=$max(\gamma_t)$;//Estimate state at time t
        Calculate the backward variable;// Eq. (4.7)
    End for
    Re-estimate model parameters, A, B, C, π //Eqs. (4.14)(4.15)(4.16)(4.18)
End for

4.4.3 Methodology

Our aim is to model the complete movement track in a way that each state in the model is either a stay-point or the transition path from one stay-point to another, where spatial coordinates have some form of spatio-temporal similarity. We assume that the sequence of places that the person visits is a Markov process with hidden states being the context ruling person's activities and the places, that a person visits, being observable two-dimensional spatial points.

A possible solution would be to consider each spatial point as an observation and use hidden semi-Markov model to find the most probable sequence of states that explain the observations. However, when a series of behaviors are repeated periodically (for example, over a day) they will be found as one super-state, while the whole super-state might be composed of smaller states. This way the desirable granularity, which is required to relate observations to concepts such as stay-points

and paths, is not provided. In fact, spatial points, which are closer to each other spatially (i.e. stay-points) or spatio-temporally (paths) are more probable to belong to the same state. Therefore, as the state duration distribution of different activities are different, simply considering each observation as a spatial point is not enough for finding the states that are explainable with human logic. The problem is better explained in Figure 4-11.

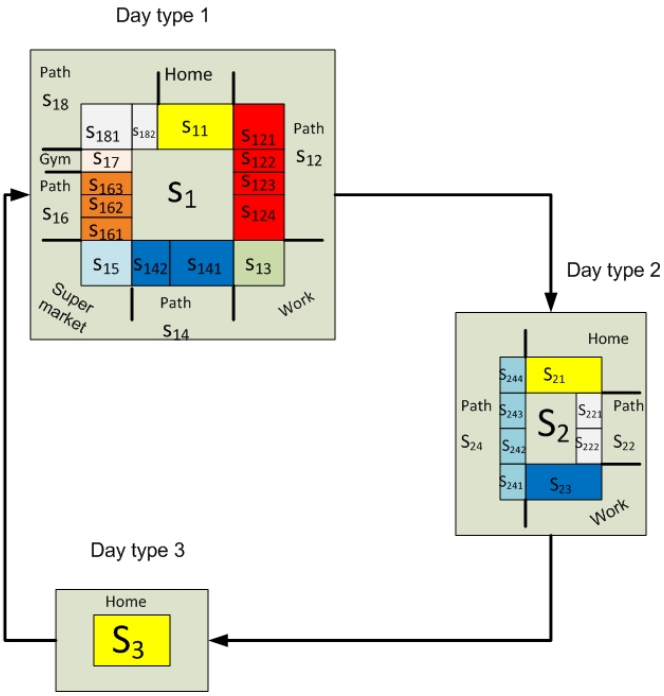


Figure 4-11 Hierarchical structure in mobility data

In this figure, the repetitive behavior of a person is illustrated. This behavior is consisted of three super-states with duration of a day. As seen, each super-state is also composed of a number of smaller states, which represent visit to different places each with their specific durations. As will be shown in Section 4.4.4.2, by using HSMM only the higher level states are found giving a very high granular view of the movement pattern where sometimes a complete day is discovered as a single state. We identified the following problem with the original HSMM:

The original HSMM treats observations as nominal values. Thereby, there is no consideration for the distance between the observations.

In order to solve the problem mentioned above, we propose using a hierarchical hidden semi-Markov model taking into account the distance between observations

in each state. This hierarchical model is defined as $\lambda = (Q, O, A, B, C, \pi)$ such that each state s_i in the model is itself a hierarchical hidden semi-Markov model $\lambda_{s_i}^h$, $(h > 1)$:

$$\lambda_{s_i}^h = (Q_{s_i}^h, O_{s_i}^h, A_{s_i}^h, B_{s_i}^h, C_{s_i}^h, \pi_{s_i}^h) \tag{4.20}$$

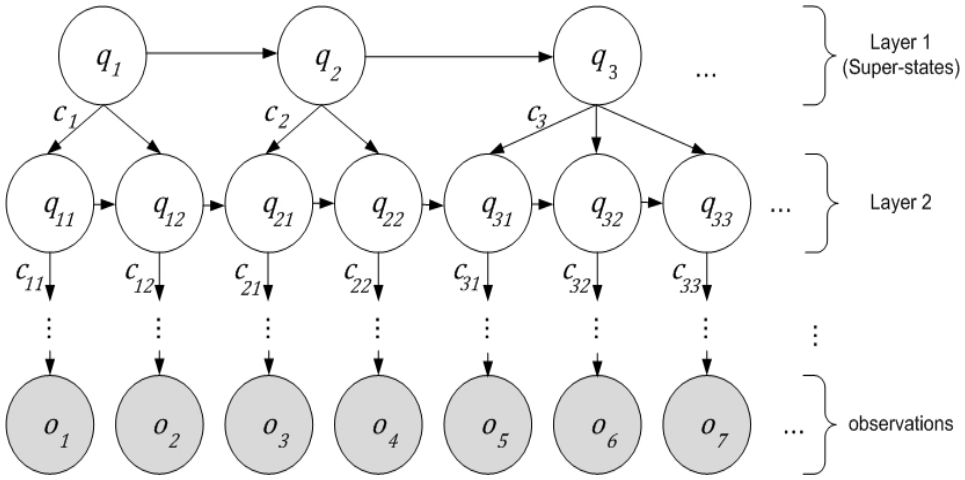


Figure 4-12 Graphical model representing hierarchical hidden semi-Markov model

The new observation sequence $O_{s_i}^h$ is only composed of the observations which were categorized into one higher-level state $O_{s_i}^h = \{o_t | q_t^{h-1} = s_i^h\}$. Each state in the final level (h) is only composed of observations, which are spatially close to each other. Figure 4-12 visualizes our proposed hierarchical hidden semi-Markov model through a graphical model.

Algorithm 4.3 summarizes the procedure of training our hierarchical model. It gets the input sequence and models the mobility pattern. Due to various environmental (such as cloud cover) and technical reasons (such as device malfunction), it is improbable that equal coordinates are reported for one place. Therefore, we firstly map location coordinates into cells of a gridded map where each observation is replaced by the relevant cell id, where it is located. The algorithm further proceeds as follows. First, hidden semi-Markov model is used to model the input sequence and to find the super-states in the model (line 1). It is probable that regular days with similar repetitive sequence of places being visited are found as one state. To have an insight with higher resolution, in case in each of these high level super-states, there are observations with a distance greater than a threshold then, that state will be chosen for being remodeled. On the next step, we apply hidden semi-

Markov model on each of these states (lines 7-9). This step can be repeated until no other states with such condition are found.

Algorithm 4.3 (HHSMM)

INPUT: $\boldsymbol{M_{1..h}}$ **(Maximum number of states in each level),** $\boldsymbol{th}$ **(distance threshold),** $\boldsymbol{D_{1..h}}$ **(Maximum state duration),** $\boldsymbol{O}$ **(observation sequence)**
OUTPUT: $\boldsymbol{A_{s_i}^h}$ **(State transition probability matrices),** $\boldsymbol{B_{s_i}^h}$ **(Emission probability matrices),** $\boldsymbol{C_{s_i}^h}$ **(State duration probability matrices),** $\boldsymbol{\pi_{s_i}^h}$ **(initial probability matrices)**

1: $[A, B, C, \pi, Q]$= TrainHSMM(M_1, D_1, O); // Train the basic level HSMM;
2: **For** $i = 2$ **to** h **do**
3: $CandidateStates$= all states found in previous level;
4: **While** $CandidateStates$ is not empty **repeat**
5: Remove all state from $CandidateStates$ with points lying within a
6: circle with radius $\boldsymbol{th}$;
7: **For** $j = 1$ **to** length ($CandidatesStates$) **do**
8: $O_{s_j}^i$={$o_t | q_t^{i-1} = s_j^{i-1}$ };
9: $[A_{s_j}^i, B_{s_j}^i, C_{s_j}^i, \pi_{s_j}^i, Q^i]$= TrainHSMM($M_i, D_i, O_{s_j}^i$);
10:
11: **End for**
12: **End while**
13: **End for**

4.4.4 Evaluation

4.4.4.1 Complexity analysis

Complexity of the light Baum-Welch training algorithm [204] is $O((MD + M^2)T)$ and the memory required for its training is $O(MT)$. Here M is the maximum number of states, D is the maximum state duration and T is the length of observations. Like all hidden Markov based algorithms, when a large number is chosen for the states and their duration (the maximum "naïve" number for M and D is the number of unique observations, and length of the observation sequence, respectively), the algorithm becomes computationally expensive. This, however, is not the case for $HHSMM$ algorithm. As shown in [185], there is high degree of temporal and spatial regularity in human trajectories, and each individual can be characterized by a significant probability of returning to a few frequently visited locations. Due to this reason, a high degree of people's activities can be summarized using very little number of super-states, which can be analyzed in more detail in case of necessity. The advantage of this hierarchical model is that its complexity is adjustable. It is

not required that the number of states are initially set equal to all unique observations. A limited number of states, with longer durations for the higher levels can be used. In each iteration of the algorithm, the number of states M_h increases while the parameters D_h and T_h decrease, leaving the learning complexity for each intermediate state in each level balanced: $O\big((M_h D_h + M_h^2)T_h\big)$, $(T_h < T_{h-1}, D_h < D_{h-1}, M_h > M_{h-1})$. Therefore, the model can be efficiently trained with respect to the resources available and the granularity required. The hierarchical model also provides the possibility of further performance improvements in terms of sampling frequency and resolution of observations in each level. In higher levels, the number of super-states is limited and sampling with low frequency is enough. By adjusting the size of the grid based on the movement area, the number of distinct observations will be reduced requiring less number of states for higher-level states.

4.4.4.2 Performance evaluation

In this section, we evaluate the performance of our proposed model and compare it with the other models in literature, using both synthetic and real datasets. It is not possible to compare this modeling technique with the one proposed in the previous section. The reason is that, by only focusing on the transitions the previous model only works for frequent trajectories, and it does not model trajectories as a whole. To be more precise, and inspired by [205], we choose the following models for performance evaluations which model trajectories as a whole:

Spatial Prior model (SP): In this model presence in each location depends on a prior location. SP is purely spatial and does not use any temporal context.

$$p^{SP}(o_t = v_i |\, t = t_i, o_{t-1} = v_j)$$

$$= p(o_t = v_i | o_{t-1} = v_j) \tag{4.21}$$

Hourly Prior model (HP): In this model presence in each location depends on its hourly visit distribution.

$$p^{HP}(o_t = v_i |\, t = t_i, o_{t-1} = v_j) \tag{4.22}$$

$$= p(o_t = v_i | t_i \bmod 24 = h)$$

Spatial-Hourly Prior model (SHP): In this model presence in each location depends on the hourly distribution, as well as the prior location:

$$p^{SHP}(o_t = v_i |\, t = t_i, o_{t-1} = v_j) \tag{4.23}$$

$$= p(o_t = v_i | o_{t-1} = v_j \,\&\, t_i \bmod 24 = h)$$

Hidden Semi-Markov Model (HSMM): This model is the basic hidden semi-Markov model where presence in each location depends on the current state, and the residual time of the states:

$$p^{HSMM}(o_t = v_i | \, t = t_i \,) \tag{4.24}$$

$$= p(o_t = v_i | \;\; (q_t, \tau_t) = (s_m, d))$$

Hierarchical Hidden Semi-Markov Model (HHSMM): This model is the one proposed in this chapter where presence in each location depends on a hierarchy of current states, and their remaining times:

$$p^{HHSMM}(o_t = v_i | \, t = t_i) \tag{4.25}$$

$$= \; p(o_t = v_i | \forall h, \; (q_t^h, \tau_t^h) = (s_m^h, d^h))$$

Three mobile entities (two people, and one capricorn) have been chosen from Dataset 2 and 3 with three different movement profiles. Table 4-4 summarizes the movement profile of each of these mobile entities. As seen, these three cases represent three general movement profiles, which are 1) high range movement, high average speed, 2) medium range movement, medium average speed, and 3) low range movement, low average speed. We see that for the second user, both maximum speed and movement area have large values. The maximum speed is in range of an airplane's speed (254 Km/h), which can also be explained by a number of coordinates in the dataset, which are in proximity of the airport.

Parameter	Mobile entity		
	Geolife User 1	Geolife User 2	Capricorn 1
Movement area (km^2)	76.6	5.2×10^5	2.8×10^3
Total disp	1.4×10^3	9.7×10^3	2.6×10^4
Dt (days)	76	254	133
Average speed (km/h)	5×10^{-5}	0.24	0.08
Max speed	71.6	240	2
Missing	76%	88%	71%

Table 4-4 Movement profile of the mobile entities

4.4.4.2.1 Model-based prediction accuracy

In order to evaluate the models, we chose to test how we can use them to accurately predict near future events. Our analysis is composed of two phases:

- **Training phase:** First, all three datasets are equally sampled per hour forming a time-series where missing values are replaced by 0. Next, we

divide each dataset into two parts. During training, the first half is completely given as input to Algorithm 4.3. The maximum state duration is 168 and 24 hours, which represent states of maximum size of a week and a day. While these values are chosen with respect to the length of datasets used for training, longer durations for super-states can be used to find longer patterns when the datasets are larger. The number of states is set to 10 and 5 for the first and second level, respectively. During tests, we observed that the number of states chosen is more than enough for all datasets, as some states are not assigned to any observation. The distance threshold used for algorithm to re-model a state is 1000 meters. After the model is trained, for $HSMM$ and $HHSMM$ models we calculate a $N \times M$ size matrix R which represents the relation between observations and states ($r_i(s_j) = p[q_t = s_j | o_t = v_i]$)). This matrix is used in prediction.

- **Prediction phase:** We check predictability of the models on the second half of the dataset. For each two consecutive timestamps where data is not missing $\{\forall\ (i, i+1) |\ (o_i, o_{i+1})\ are\ not\ missing\}$, and o_{i+1} had been observed in the training dataset, we check to see how we can predict the data of the second timestamp (o_{i+1}) from the prior one (o_i).

The procedure was repeated 50 times with different grid sizes (varying from 10×10-500×500 for the first user, 500×500-1000×1000 for the second, and 1×1-50×50 for the capricorn). These sizes have been chosen based on the movement ranges. Figures 4-13, 4-14, and 4-15 show a comparison between the efficiency of each of these models in terms of their prediction accuracy. Figure 4-13 and Figure 4-14 represent the results of performing experiments on the users from Geolife dataset (Dataset 3) and Figure 4-15 is that of Capricorn data (Dataset 2). In each figure, the movement range of the mobile entity after being sampled, total prediction accuracy, prediction of change accuracy, and the cost of wrong prediction are shown. These parameters are explained below:

- **Total prediction accuracy:** This graph represents the total correct predictions both when the next destination is in the same cell and when it is in another cell.
- **Prediction of change accuracy:** As the periods of stay and movement are unequal, it is almost always easier to predict points where the mobile entity is stable (predicting the current spatial point as the next destination ($o\ (t+1) = o\ (t)$)). Therefore, as well as showing the accuracy of models in terms of total prediction, we also show the results of predicting the points which represent a change from the previous timestamp ($o\ (t+1) \neq o\ (t)$). This helps in showing the difference of algorithms in predicting these two different types of measurements.
- **Cost of wrong prediction:** This graph represents the number of cells proposed with highest probability for each wrong prediction. This will

show the cost of each wrong prediction. The reason for showing this graph is that, for *HSMM* and *HHSMM*, it is possible that each state is composed of a group of observations. Therefore, by using the observation/state matrix (*R*) this group of points, belonging to one state, will be suggested as the next point prediction having the same probability ranges. In the other models, however, the most probable point has a higher probability, which can be used in prediction. In order to be fair, we also compare the methods in terms of the cost of this inaccuracy. As the cost of HHSMM is lower than HSMM, we adjusted the cost of the other models with this model by accepting more predictions. This way, for the other models we always accept the top 5 most probable points for predicting the next destination.

Looking at Figure 4-13, Figure 4-14, and Figure 4-15 one notices the following:

For the first two datasets, the *HSMM* model performs considerably better than all the other models in terms of total and prediction of change accuracy. This comes, however, with a considerable high cost for each wrong prediction. This represents the high granularity of the states, which is the outcome of *HSMM*. *HHSMM* follows *HSMM* in total and prediction of change accuracy with a cost of wrong prediction being much lower than that of *HSMM* and in range of the other methods. Prediction of change accuracy with these two methods is higher than the others. This is resulted by correct duration estimation for each state. For the Capricorn dataset, *HSMM* and *HHSMM* are very close in prediction of change accuracy whereas the total prediction accuracy of *HSMM* is higher than that of *HHSMM*. However, this time the cost of wrong prediction of two methods is very close. This is explained by the fact that, the animal's movement is less structured, and that the hierarchical structure has not been able to add to the accuracy. In this case, the higher granular model is more successful.

In most cases by increasing the grid size, the accuracy decreases. This is due to an increase in the number of unique observations. The uneven shape of lines is due to discretizing observations to grid cells and is caused by the fact that the data had not been preprocessed. The other reason is that for *HSMM* and *HHSMM* the result of training is not always a unique model. Therefore, the best model is chosen (empirically) after ten times of training.

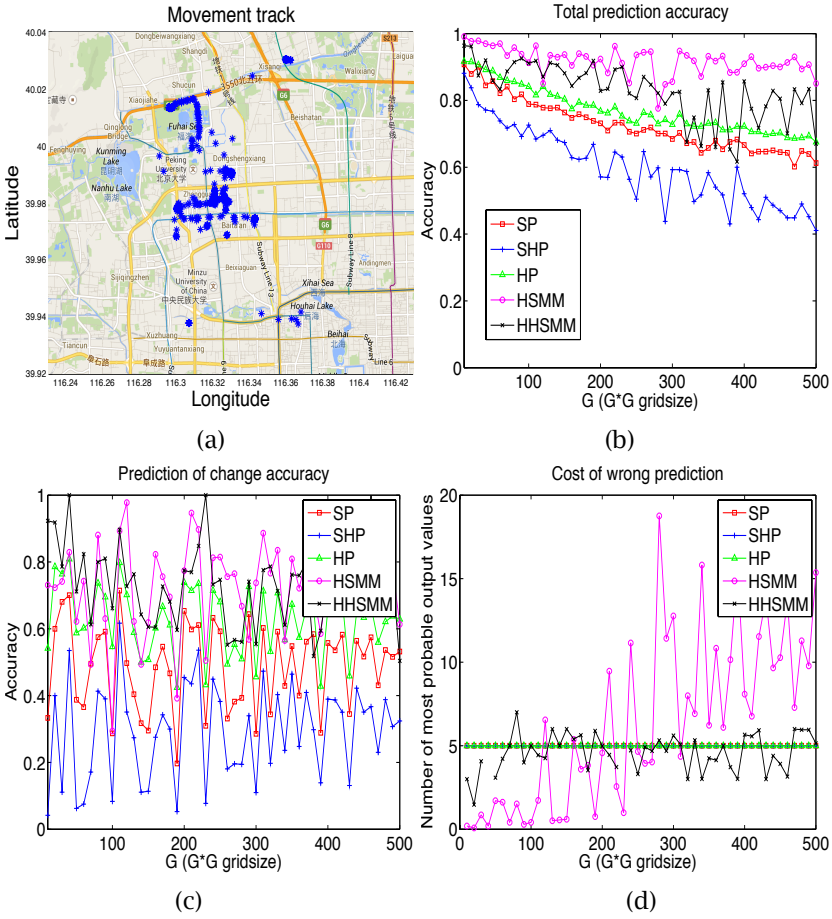


Figure 4-13 Applying different movement modeling techniques on the first user of Dataset 3, (a) Original movment tracks, (b) Total prediction accuracy, (c) Prediction of change accuracy, (d) Cost of wrong prediction

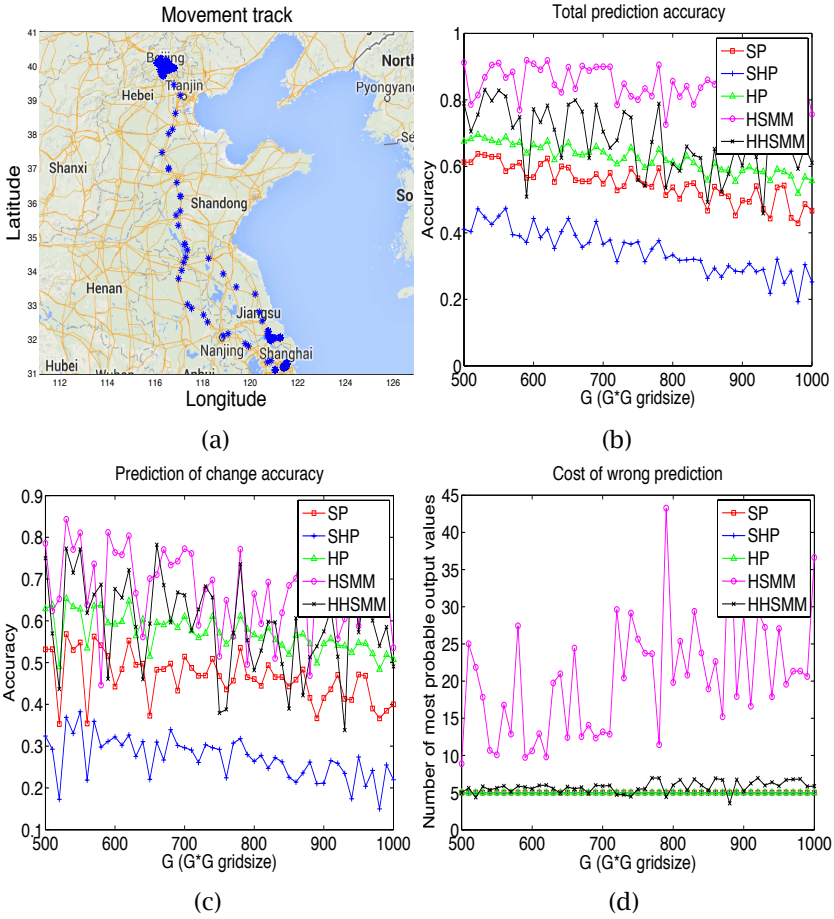


Figure 4-14 Applying different movement modeling techniques on the second user of Dataset 3, (a) Original movment tracks, (b) Total prediction accuracy, (c) Prediction of change accuracy, (d) Cost of wrong prediction

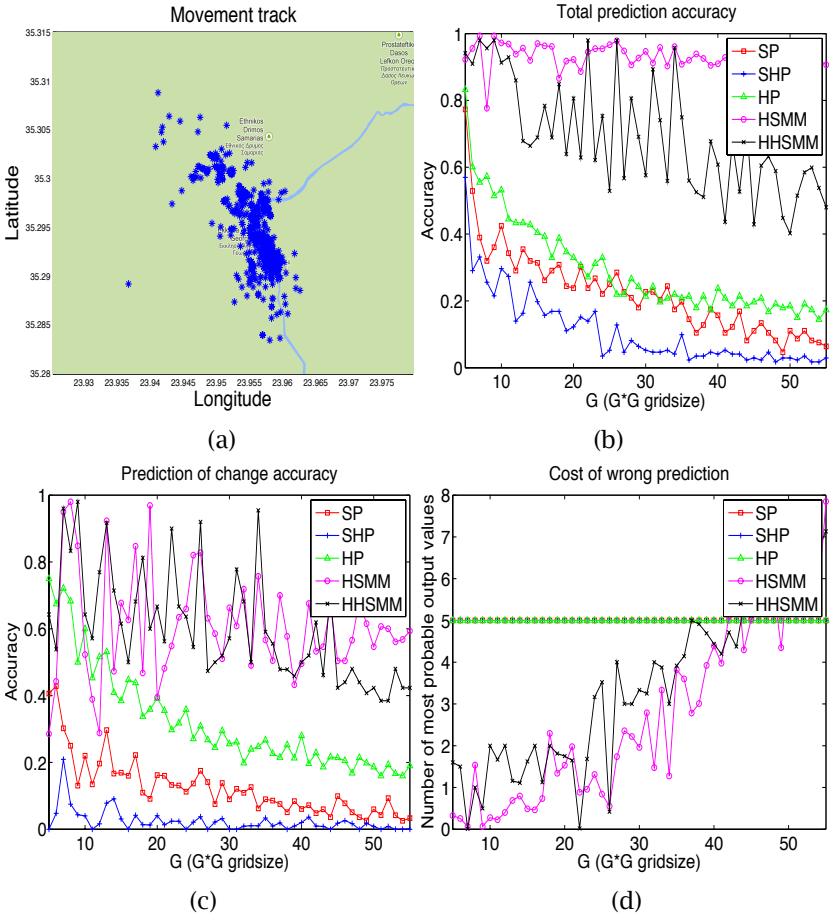


Figure 4-15 Applying different movement modeling techniques on the first capricorn of Dataset 2, (a) Original movment tracks, (b) Total prediction accuracy, (c) Prediction of change accuracy, (d) Cost of wrong prediction

4.4.4.2.2 Robustness against noise and missing values

In this section, we prove the validity of our approach in presence of noise and missing values. For this purpose we use a synthetic dataset. This test helps us in checking the sensitivity of the models in a controlled setting. In order to produce the synthetic data, a movement generator was written with the parameters mentioned in Table 4-5 to produce a test sequence. It should be noted that, the value r is represented in terms of the offset added to the raw longitude and latitude.

Parameter	Value
σ_{start}	120 min
σ_{end}	120 min
r	0.001
L	10
K	8
Missing samples (θ)	5-50% (N×24)
Noise (ρ)	5-50% (N×24)
Number of Grid cells	100×100
Total number of paths and places	7

Table 4-5 Parameters chosen for the test with synthetic dataset

The test sequence produced with the movement generator is composed of the repetition of a sequence of geo-spatial points, which can represent a repetitive behavior of a person in visiting a number of places ($test_i = \{x_i, y_i | i \in [1, L{\times}24]\}$). K number of places and the paths connecting them are chosen. The event of start and end of visit to each of these places is expected to be at t_{start} and t_{end} and the actual visit happens within $t_{start} \pm \sigma_{start}$ and $t_{end} \pm \sigma_{end}$. After forming this sequence we perform the following tests to analyze the effect of missing samples and noise:

- **Test 1 (Missing samples):** We generate θ random indexes and replace the indexing values (x_θ, y_θ) by $(0, 0)$ (representing missing observations). Next, we train each of the models on the resulting sequence. The success of each model is in correctly finding observations, which can replace each missing value.
- **Test 2 (Noise):** We generate ρ random indexes and replace (x_ρ, y_ρ) with a noisy value ($x_\rho + e_x$, $y_\rho + e_y$) where e_x and e_y are randomly chosen from $[0, r]$.

The success of each model is on correctly replacing the noisy observation with the original value.

For the models *SP*, *SHP*, and *HP*, we chose the values, which had a probability over 0.5 for predicting the missing and noisy values. For *HSMM* and *HHSMM* we chose the cells, which belonged to the state detected with probability more than 0.5.

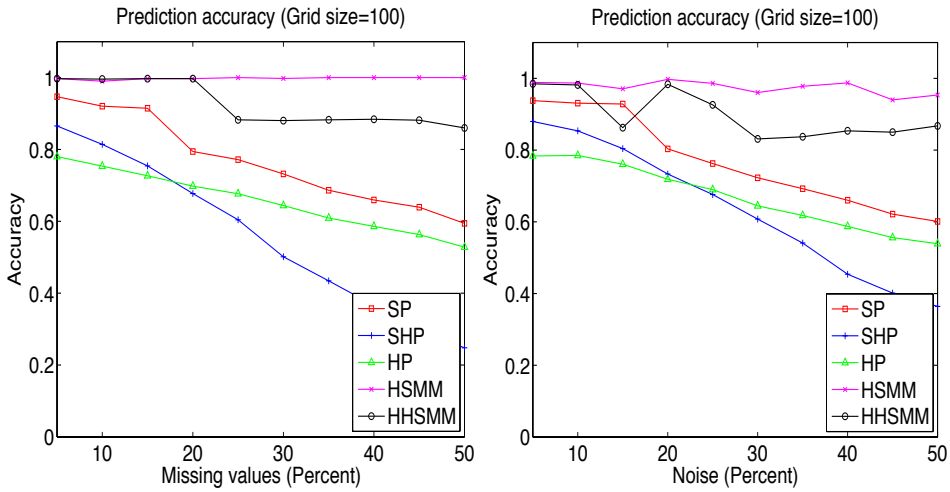


Figure 4-16 Success of algorithms in predicting the (a) missing value and (b) noise

As seen from Figure 4-16.a and Figure 4-16.b, *HSMM* followed by *HHSMM* are superior to the other models both when missing values and noise are present in the dataset. Even when noise or missing values reach up to 50%, these two models perform considerably well. This is thanks to considering both forward and backward variables, which are able to find the best model representing the entire dataset. The accuracy of *HSMM* is higher than *HHSMM* as it predicts all the points belonging to the super-states whereas, *HHSMM* gives finer grained predictions having a slightly reduced accuracy.

4.4.4.3 Case studies

In order to show the process of training *HHSMM* and the form of patterns discovered by it, we show the procedure of building a two-level hierarchical model with Algorithm 4.3 on three Datasets 1,2 and 3.

4.4.4.3.1 Case study using Dataset 1

Figure 4-17.a shows that the mobility data of User 1 is more concentrated in a small area. In order to train *HHSMM*, we chose the value 168 for the maximum state

duration in the first level, the values 60 and 24 for the maximum state duration in the second level, and 10 for the number of states in each level.

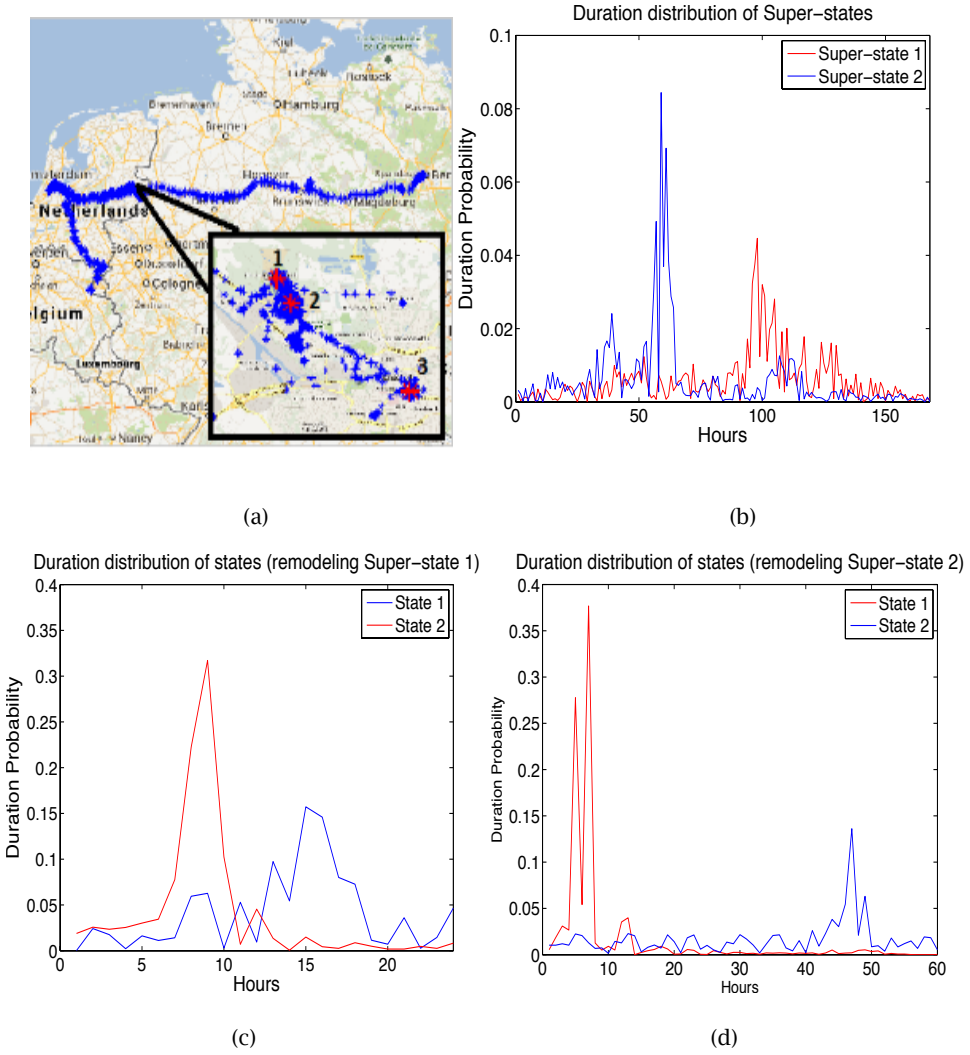


Figure 4-17 Modeling the mobility data of user 1 in Geolife dataset (a) duration distribution of superstates, (b) remodeling super-state 1 (weekdays), (c) remodeling super state 2 (weekends).

In the first level, two super-states are found with different duration distributions shown in Figure 4-17.b (while we chose number 10 for the number of states, after

training, the rest of the states were not assigned to any observation). These two duration distributions, with means near 100 and 60 hours, evidently represent the general distinction in mobility behavior of this person in weekdays (4.5 days) and weekends (2.5 days). This is an interesting positive characteristic of our model, as it can find proper duration distribution without us making any assumption on this typical weekly behavior. Such patterns were previously found with complex periodicity analysis, which was the subject of Chapter 3.

After remodeling, for both super-states 1 (weekdays) and 2 (weekends), 2 lower-level states are found. The duration distributions of these states are shown in Figure 4-17.c-d. During weekdays (Figure 4-17.c) there are two states with duration of 8 and 16 hours, respectively. These two can represent the stays at work and home of this user (spatial neighborhood 1 and 2 in Figure 4-17.a). During weekends (Figure 4-17.d) there are two stays of near 50 and 5 hours, which represent stays at home and market (spatial neighborhood 1 and 3 in Figure 4-17.a). These results are also inline with the results achieved in Section 3.6.1 by extracting periodic behavior of this user.

It should also be mentioned that, without a rough guess about the emission and transition matrices, hidden semi-Markov models do not always converge to the same results. In order to have understandable states, we repeat the learning process few times to get state durations, which follow a normal distribution.

4.4.4.3.2 Case study using Dataset 2

In section 4.3.3.4, we saw that the mobility track of one of the capricorns (Entity 1) was more versatile in term of stay-points and transitions. As mentioned before, there are constantly 8 missing samples in this dataset. Removing the constant missing samples we acquired a time series where each 16 timestamps represent 1 day.

We hierarchically modeled the mobility data of this entity using Algorithm 4.3 (HHSMM). The results are presented in Figure 4-18 and Figure 4-19. After modeling the movement for the first time, with maximum duration of the length of total dataset, we found three general super-states. The duration distribution of these super-states are shown in Figure 4-18.a. As suggested by the model, these three super-states can summarize three different long epochs where the animal has different behavior. More specifically, the model has been able to find changes in the mobility behavior. For instance, as seen in this figure, the first super-state has a duration of 1500 timestamps (1500/16=93 days) while the other two have durations less than 500 timestamps.

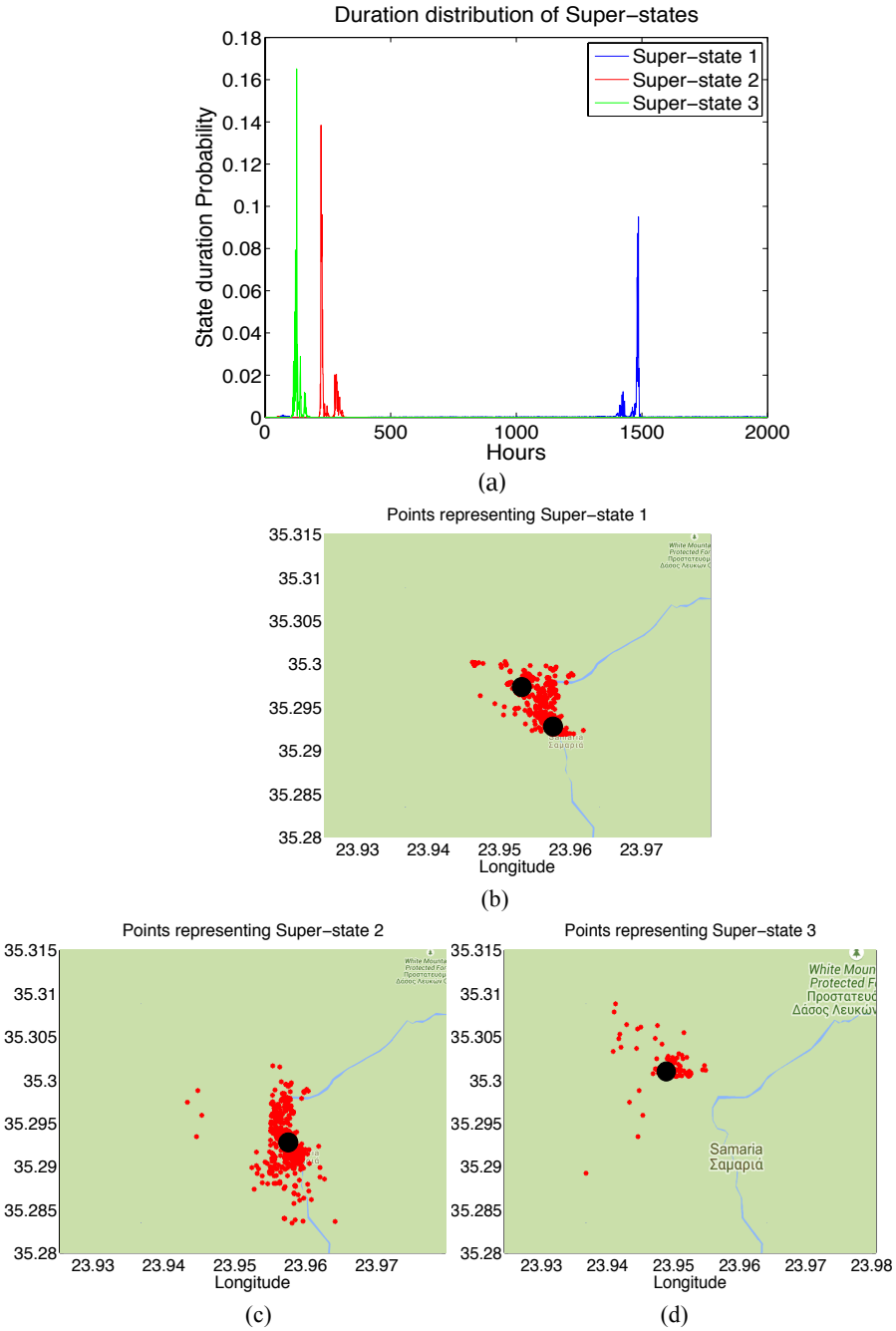


Figure 4-18 Modeling the mobility data of a Capricorm, (a) duration distribution of superstates, (b-d) observations representing each super-state

The observations representing each super-state are shown in Figure 4-18.(b-d) in red color. The black circles on each map show the observations that highly represent that state ($p\,[o_t = v_i | q_t = s_j]$). As suggested by Figure 4-18.b, the first super-state shows the epoch where the animal has mainly spent his time in two different stay-points (black circles). The second and third super-states represent days where the animal has mainly stayed around a limited area shown in Figure 4-18 (c-d). We remodeled the first super-state, which was represented in Figure 4-18.b, and found two other smaller states with their specific duration distribution. The results are shown in Figure 4-19.a-b. The observations, which highly present each state, are depicted with black circles on the map. The duration distribution of each state shows stays of 6 and 10 timestamps, respectively. Summing the duration distribution of these two states we acquire 16 which is the number time-stamps per day.

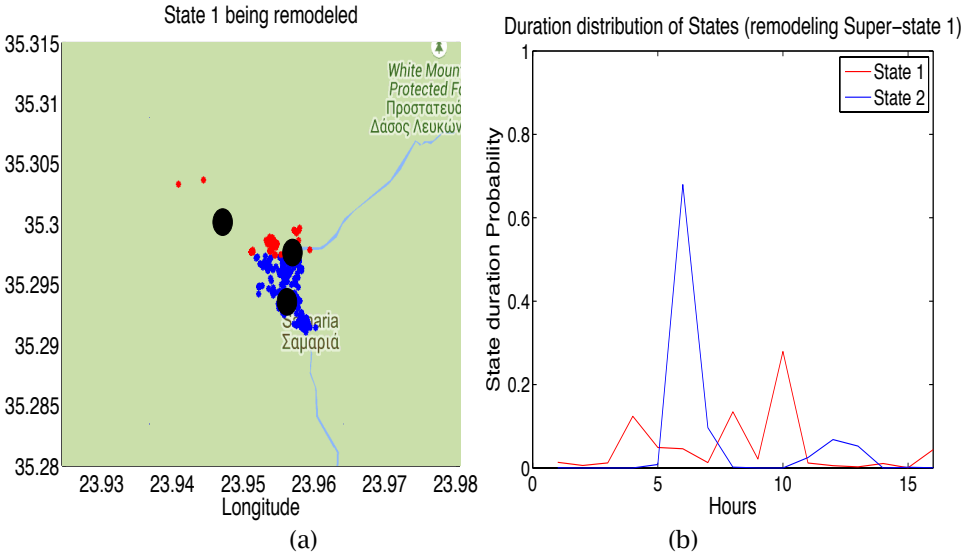
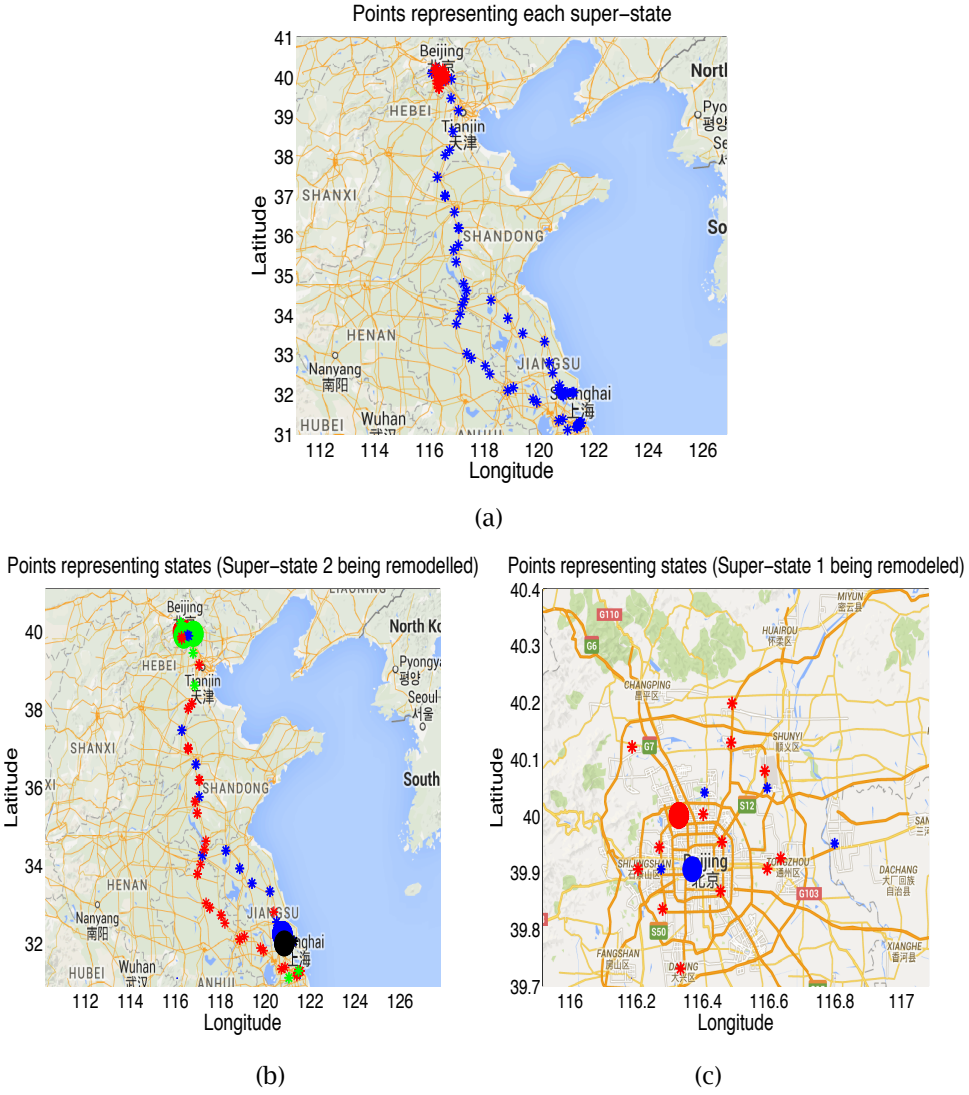


Figure 4-19 (a) Observations representing the states found by remodeling super-state 1, (b) duration distribution of the two states found by remodeling Super-state 1.

4.4.4.3.3 Case study using Dataset 3

As seen in Figure 4-20.a, the mobility data of User 2 of Geo-life dataset is composed of very long travel sequences. We used the Algorithm 4.3 (*HHSMM*) with a grid size of 100×100, values 168, and 24 for maximum state duration of first and second level, and 10 for the number of states in each level.



(a)

(b)

(c)

Figure 4-20 Modeling the mobility data of user 2 in Geolife dataset, (a) Superstates, (b,c) Second level states in each super-state.

We were able to find two general super-states for this user as shown in Figure 4-20.a. The super-state colored in blue represents the points corresponding to long traveling sequences. Due to its rare nature (9% of dataset) and average high speed of the user's movement in this super-state, most of the points in this super-state are only observed once (the median and mean of the number of times each

observation is observed in the dataset is 1 and 3, respectively). However, as seen in Figure 4-20.b, after the observations in this super-state are remodeled, 4 states are found which can represent the ways to and from two stay-points, as well as the stay-points in different cities. Although most of the points in this super-state (blue) are only observed once, the similarity between points in this state relates to the fact that they are followed by visit to the points in the other state. This way the model also works as an abnormal pattern detection method. Existing Markov model-based methods, which only use stay-points for modeling, are unable to find such visible states. The red super-state is a dense representation of points in an area where more than 91% of the observations are located. After the super-state is remodeled (Figure 4-20.c), two lower-level states are found.

4.5 Comparison

In Sections 4.3 and 4.4, we elaborated our two proposed techniques for trajectory modeling. Each of these techniques has their own pros and cons. It is worthwhile to take the differences into account, before using them for a specific application. In what follows, we name a number of parameters important for modeling techniques and compare the two trajectory modeling algorithms (Algorithm 4.1 and 4.3) proposed in this chapter.

Capability of pattern discovery: Generally, state-space models are extremely powerful in terms of pattern discovery. Such capability is provided through expectation maximization in estimating model parameters. There is no prior description of the model or the pattern to look for. Any pattern is discoverable, as long as it can be expressed in terms of states with specific duration and their transitions. Therefore, such techniques cover a family of spatio-temporal patterns rather than a specific pattern. As seen in Section 4.4.4.3, using *HHSMM* we were able to discover three different types of patterns (Periodic, abnormal, and change). *SPdefine*, however, is only capable in finding one form of pattern, which is the most frequent pattern.

Dependence on input parameters: Both of the algorithms proposed in this chapter require certain input parameters. Algorithm 4.3 (*HHSMM*) requires the maximum number of states and maximum state duration. Algorithm 4.1 (*SPdefine*) requires the mean threshold for the number of points in each grid cell and the minimum stay duration in stay-points. In order to choose the parameters of *HHSMM* one can choose the highest value possible. This way only the memory requirement of the algorithm is increased. However, *SPdefine* strongly depends on parameters such that, careless choice of parameters jeopardizes the accuracy of the algorithm or even causes the algorithm not to perform.

Easy interpretation of results: *SPdefine* is designed based on a known concept in mind (matching trajectories to paths). Therefore, the results achieved by it require

no further investigation. This way, they are straightforward to be used in any application. The fact that *HHSMM* is more general than specific, also makes its interpretation challenging. After modeling with *HHSMM*, one may need to spend some time interpreting the pattern by looking at duration distributions and state-transitions probabilities.

Guaranteed convergence: *Spdefine* always provides the same results. However, this is not the case for *HHSMM*. There is no guarantee that the parameter learning procedure of state-space models converge to the most understandable model. In other words, you do not always get the same model when you run the algorithm.

Reasonable memory and processing: As we explained in the complexity analysis in Section 4.4.4.1 and 4.3.3.1, *SPdefin* consumes less memory compared to *HHSMM*. However, when the model is learnt, classification based on both of these approaches is reasonable both in terms of memory and processing resources.

Independence of preprocessing phase: *SPdefine* requires pre-processing to extract stay-points and removing noisy samples. However, *HHSMM* is directly applied on the data points after they are gridded.

Modeling the whole trajectory: As mentioned before, *SPdefine* is only applicable on trajectory segments, which represent transitions from one stay-point to another. Whereas, *HHSMM* is applicable on complete trajectories. Specifically, this probabilistic approach is powerful in capturing the duration distribution of stays in stay-points.

Resistance against missing values: *HHSMM* is implicitly resistant against missing values. Learning the parameters is by considering both forward and backward transition between states and then iteratively finding the best set of parameters for the model. This way, the support for missing values is extremely strong. Even when data is continuously missing for a long duration, the parameter estimation still finds the best model. However, in *SPdefine*, the solution for missing values (using collective knowledge) works as long as, values are missing for a limited duration.

Implicit resistance against Noise: *SPdefine* can cope with noisy measurements as long as they are within a certain threshold. *HHSMM,* on the other hand, has an implicit resistance against noise. Using the emission probability table, the importance of observations is defined based on their probability. As noisy measurements have lower probability, they will not degrade the parameter learning performance.

Property		SPdefine	HHSMM
Capability of pattern discovery		-	+
Dependence on input parameters		-	+
Easy interpretation of results		+	-
Guaranteed convergence		+	-
Reasonable Memory required	**Learning**	+	-
	Inference	+	+
Independence of preprocessing phase		-	+
Modeling the whole trajectory		-	+
Resistance against missing values		+ (Short term)	+ (Long term)
Implicit resistance against noise		-	+

Table 4-6 Comparison of the probabilistic and determinisitic modeling algorithms proposed in this chapter

Table 4-6, summarizes the comparison of these two algorithms in terms of the above-mentioned parameters.

4.6 Summary

In this chapter, we proposed two techniques for trajectory modeling. The first approach was based on a hierarchical clustering algorithm. In this algorithm, we segment trajectories to the smallest meaningful unit of movement and then find the frequent segments. In our second approach, we addressed the problem through proposing a hierarchical modeling technique. We used hidden semi-Markov models to model the trajectories, considering movement in different contexts, as observations that are repeated with certain durations. The first approach is more powerful in finding only one specific form of patterns. As seen before, in order to make a complete deterministic model we need to make many assumptions about the patterns, and thresholds. The second approach is more powerful in discovering general and longer-term patterns. As shown with examples from real movement datasets, using this technique we were able to find patterns (periodic, rare, frequent) without having any presumption of their existence. In spite of the power of the second approach in discovering patterns, it still requires supervision during the learning process.

With respect to the abovementioned, it is suggested that, the second technique is used initially when not enough knowledge is available on the trajectories to find

the general structure of patterns. The first approach can be used afterwards on each specific form of pattern discovered before.

Chapter 5

Social context mining from mobility data[8]

Large volumes of mobility data not only provide information about individuals themselves, but also about their interactions with each other. Unlike mining individual mobile entity behavior, discovery of social ties and interactions using mobility data has not yet been fully explored. Compared to data types such as phone call and message posts, mobility data convey less information about the direct interaction between entities. Therefore, identifying the type of tie between two entities by only using mobility data is a great challenge. In this chapter, we propose a method for identifying the type of social tie between mobile entities by looking at the spatio-temporal correlations at stay-points based on the purpose of visit to them. To this end, we propose two types of indicators based on mutual information for identifying the purpose of visit to different locations and relate it to the social tie between entities. Our experimental results show that, compared to the popularly used co-location indicator, these indicators can better represent the strength of social ties between mobile entities.

5.1 Introduction

In previous chapters, we studied spatio-temporal trajectories from an individual mobile object's point of view. In this chapter, we look at the possibility of extracting social context from mobility data. An interesting topic in studying such data is finding the existence of a social tie between entities and describing the purpose of such ties being formed. Social ties between mobile entities are formed due to having different relationships. In case of humans, these ties are formed due to friendship, work-related acquaintance, and family membership, to name but a few. Each type of social tie conveys different information about the habits, interactions, and information exchange between the entities connected by it. While friends tend to show more similarity in their interests and habits, more information is exchanged between acquainted people [207]. Distinguishing between

[8] This chapter is partly based on:

[206] M. Baratchi, N. Meratnia, and P. J. M. Havinga, "On the use of mobility data for discovery and description of social ties," in *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM'13)*, Niagara Falls, Canada, 2013, pp. 1229-1236.

different types of social ties is essential for discovering different communities and understanding the interaction between individuals [208]. In order to find communities with a certain characteristic, identifying members, who are connected with a social tie relevant to that characteristic, is important. Nowadays, this information can also be used in different recommendation systems [6]. Similar to human studies, differentiating social ties is important in ecological research. Being able to differentiate social ties between animals can provide insights about their evolution and gene flow, maintenance of society, analyzing epidemic patterns, transmission of information, events, and social learning [209].

The success and reliability of a system for analyzing social ties greatly depends on the data it uses. The types of data, which are normally used for the discovery of social ties, are acquired from emails, phone calls, and online social networks. Such data are richer than mobility data as they convey more interaction information. Discovery of social ties using solely mobility data is challenging due to their lack of interaction content. For example, working in the same building does not guarantee that two people are friends or even acquainted (example: those who work in two different floors of a building or in two different departments of the same organization). In contrast, the fact that two people post on each other's Facebook wall, send an SMS or email to each other, or talk on the phone indicates existence of an interaction between them. Furthermore, uncertainty of mobility data acquired by existing technologies, which may be in order of tens of meters, makes the discovery of social ties even more difficult.

5.1.1 Contributions

Motivated by the fact that entities with social ties, to some degree, share spatio-temporal context [210, 211], our contribution in this chapter can be summarized as:

- Using mobility data for identifying social ties between mobile entities with daily behaviors of different entropies.
- Proposing two information theory-based indicators to measure the correlation between mobile entities at stay-points based on their purpose of visit to them.
- Identifying the nature of social ties between two mobile entities based on the above-mentioned indicators.
- Experimenting with Nokia Mobile Data Challenge dataset to compare the proposed indicators with normal co-location indicator on representing the strength of social tie achieved from phone call/sms features.

The rest of this chapter is organized as follows. Section 5.2 and 5.3 present the related work and problem definition. The detailed description of our approach is

provided in Section 5.4. Evaluation results and case studies are reported in Section 5.5 and 5.6, while Section 5.7 is the summary.

5.2 Related work

Most of the research performed previously in analyzing social ties defines binary associations (existence versus absence of social tie) between social entities [212-215]. These researches ignore the importance of the type of social tie between entities. There are a number of previous work with the focus on link description based on the data acquired from online social networking websites [216-218], and heterogeneous networks [219]. The description and prediction of social ties in these works are normally based on a number of links formed previously by user input. In contrast, no prior information on links is available when mobility data is used. Furthermore, these works benefit from the amount of different type of "interaction" content available for each individual (number of photos tagged, number of wall posts, etc.).

More recently, identifying social ties using mobility data has been proposed. In the research presented in [220], existence of social tie between two people is inferred from the semantic similarity of their trajectories without interpreting the type of social tie. In [221], authors have used communication and mobility data from mobile phone records for finding friendships. They have used four factors, i.e., (i) presence on campus/off campus, (ii) daytime/nights, (iii) weekend proximity, and (iv) phone communication for measuring the social ties. This approach is specific to social ties in one affiliation and does not work for mobile entities with different spatial domain. Furthermore, not all people have the same working habits that are dependent on the day of week. A number of co-location metrics are introduced in [222] to be used along with mobile phone data to measure the social tie strength between people. These co-location metrics are based on the probability of two people being in the same place. The use of mobile phone data, as used in [221, 222], can bring additional interaction content to the analysis process.

In contrast to the above-mentioned research, we consider extracting social information only using mobility data. The major difference between our work and previous research in differentiating social ties [216-219] is the way we describe the links. Existing works relate the strength of the tie to the strength of friendship. Thereby, strong ties show strong friendships and weak ties show acquaintance. We however, make a clear distinction between different classes of social ties, namely friends, acquaintances, and families by analyzing two indicators. Another major difference between our work and the existing solutions is that, all existing solutions focus on the value of joint probability (co-location) of two people visiting places for measuring the strength of their social tie. The joint probability of two people in visiting one place might be higher for acquaintances who work together

than for friends who have different working and living habits. Therefore, this measure is not a good indicator for the type of social ties. We consider the use of mutual information content of people over places with both high and low frequency of visits. To the best of our knowledge, there are only two previous research [223, 224], which have considered using the mutual information content in measuring social ties between people. Authors of [224] have used mutual information to measure social ties and use it as an additional tool for prediction of human mobility pattern. The authors only use the data of people with strongest mutual information to increase their prediction efficiency. In the work presented in [223] mutual information is used to measure the social tie strength in bi-partite networks. There are, however, two major differences between our work and the work presented in [223]. Firstly, the work presented in [223] does not make any distinction between the type of ties, while we propose two indicators to describe different classes of ties. Secondly, [223] considers measuring the social tie between people using a non-location dataset of people who participated in selected one-time events. As will be shown in Section 5.5.1, such a metric is not applicable in inferring social tie information of mobile entities from their location data.

5.3 Problem Definition

Let $D = \{P_1, P_2 \ldots P_N\}$ represent a set of mobility data collected from N number of mobile entities. For each entity i, there exists a list of time-stamped measurements denoted by $Pi = \{Ts_1, Ts_2, \ldots, Ts_m\}$ over observation duration of m time stamps, where Ts_k is a two-dimensional spatial coordinate.

We are interested in inferring different classes of social ties between entities from D. These social ties can identify the nature of interaction between entities. For instance, for two people denoted by i and j, we define these social ties to be *acquaintance*, *friendship (ordinary or buddy)*, *cohabitee,* and *un-related*. In a special case, we look for known social ties between mobile entities. Acquainted are those who know each other due to an un-emotional reason. The social tie between colleagues is an example of this type. Friends have special emotional relationship. Ordinary friends only have emotional relationships while buddies have both un-emotional reasons (for instance, they work or study in the same place) and emotional ones. Cohabitees refer to people who live in the same place. There are also people who do not fall under any of these categories and have no relationships with each other. Different social ties based on the same concept may exist for animals.

5.4 Methodology

5.4.1 Background

5.4.1.1 Social ties and stay-points

A trajectory is composed of transition lines and transition endpoints. Transition endpoints are places that mobile entities stay for a considerable amount of time while transition lines are the paths, which the mobile entities traverse to reach one transition endpoint from another. Most of people's social ties are formed in places where they stay rather than on paths they traverse to get to those places. Inspired by this observation, we only use the mobility data in transition endpoints (stay-points) for describing the type of social tie between people. Based on the Theory of Homophily [225], people tend to build social ties with whom they have more similarity. Therefore, correlation and similarity of people in visiting stay-points can be used to describe their social tie.

In this chapter, we use mutual information for measuring the correlation between people at stay-points. In what follows, we first present background information on mutual information.

5.4.1.2 Mutual information

Information theory-based measures relate the information content of events to their probability of occurrence. The mutual information metric [226] (MI), measures the dependency of two random variables on each other in terms of the amount of information they share. Given two random variables X and Y, with marginal probability mass function denoted by $p(x)$ and $p(y)$, and the joint probability mass function of $p(x,y)$, their mutual information is defined as the relative entropy between the joint distribution and their product distribution $p(x)p(y)$ as stated below [226]:

$$MI(X,Y) = \sum_{x_i \in X, y_j \in Y} p(x_i, y_j) log \frac{p(x_i, y_j)}{p(x_i).p(y_j)} \tag{5.1}$$

The unit of mutual information is Bits and if two random variables are independent of each other, their mutual information is equal to 0 bits.

An extension of mutual information is normalized mutual information [227] (NMI), which scales the above-mentioned measure between 0 and 1 where $H(X)$ and $H(Y)$

are the entropy of X and Y, respectively. Normalized mutual information is calculated as follows:

$$NMI(X,Y) = \frac{2MI(X,Y)}{H(X) + H(Y)} \tag{5.2}$$

This measure shows how predictable one random variable is from another and its advantage is quantifying the information content of events by their probability. Thereby, an event which is less likely to happen, contains more information than the one, which is more likely. This property can be used in distinguishing different types of social ties. A short visit of two friends should bring more information about their social tie than a frequent visit of two colleagues at work. Mutual information metric is extensively studied in different domains of science such as biology [228]. However, its potential to identify the social tie between people from mobility data has not yet been fully explored.

5.4.2 A naïve approach for using mutual information

In this section, we explain how to utilize the mutual information metric to distinguish between people's social ties.

If we assume that mutual information can be used for measuring similarity between two people i and j, then a naïve idea will be to first compose an ordered list of time-stamped stay-points over a period of m timestamps for each person ($SPL = \{x_1 \dots x_m\}$ where x_b is SP_a when the person is at stay-point a at timestamp b, and $SP_1 \dots SP_k$ represent k different extracted stay-points). Next, we can apply normalized mutual information on the defined ordered list of time-stamped stay-points (SPL_i, SPL_j) and then take the measured value as an indicator of strength of their social tie. Although simple, the naïve approach suffers from a number of shortcomings highlighted below using Example 1:

Example 1: Let us consider four persons, i.e., Alice, Bob, Chuck, and Linda. Alice and Bob are friends. Bob and Chuck are colleagues and work in the same building. Linda is Chuck's wife and they live together. Every 8 hours, we collect data from places that these four people visit for a period of three weeks. Let us consider the activity of visiting places as listed in Table 5-1. These four persons go to work every weekday, one weekend Alice and Bob go to a musical and the next weekend Chuck and Linda go to the same musical. We give an identifier to each visited place (see Table 5-1) and represent the list of time-stamped stay-points in Table 5-2. Table 5-3 shows mutual information measured between these four persons on the set of visited places.

From this simple example (looking at Table 5-2), we can conclude that although the normalized mutual information can say how predictable behavior of a person is using information of another person, it does not well indicate how people are socially connected. Firstly, all these people have relatively high normalized mutual information with each other while they have different social ties. Furthermore, there is no distinction between the two pairs 'Alice-Bob' and 'Chuck-Linda'. The first pair are friends who only visited each other once, while the second pair live together. Also, Alice has not visited Chuck and Linda in any place but her normalized mutual information with them is as high as the normalized mutual information between Bob and Chuck, who work together.

A disadvantage of this measure is that it does not consider the fact that social tie between people is (mainly) formed due to their co-existence in the same place. The fact that people follow similar daily patterns in distinctive places cause their normalized mutual information to be high. Perhaps this is one of the reasons why normalized mutual information metric has not yet been fully explored in describing the social ties.

Place	Place Code
Alice's house	1
Alice's office	2
Bob's house	3
Bob and chuck's office	4
Musical	5
Chuck and Linda's house	6
Linda's office	7

Table 5-1 List of places

Person	String
Alice	12112112112112111511112112112112112111111112112112112112111111111
Bob	34334334334334333533334334334334334333333334334334334334333333333
Chuck	64664664664664666666664664664664664656666664664664664666666666
Linda	67667667667667666666667667667667667656666667667667667667666666666

Table 5-2 Ordered list of stay-points during a period of 3 weeks collected every 8 hours

Another drawback of the naïve approach is that it computes the normalized mutual information using the entire set of visited places without considering the purpose of these visits. To this end, it is not logical to use predictability of people over a long period of time (their entire life span) to measure their social ties.

NMI	Alice	Bob	Chuck	Linda
Alice		1	0.87	0.87
Bob			0.87	0.87
Chuck				1
Linda				

Table 5-3 Normalized Mutual information (NMI) computed using equation (5-2) measured over the set of visited places shown in Table 5-2

5.4.3 A heuristic based approach

Having highlighted the disadvantages of the naïve approach, in what follows we present our proposed heuristic approach. Our approach exploits the advantage offered by normalized mutual information measure and at the same time deals with the two above-mentioned drawbacks.

Figure 5-1 abstractly shows the process of extracting the type of social tie between two people by this approach. The core of this approach is two indicators that represent the interest in common places (*IPL*) and interest in person (*IPR*).

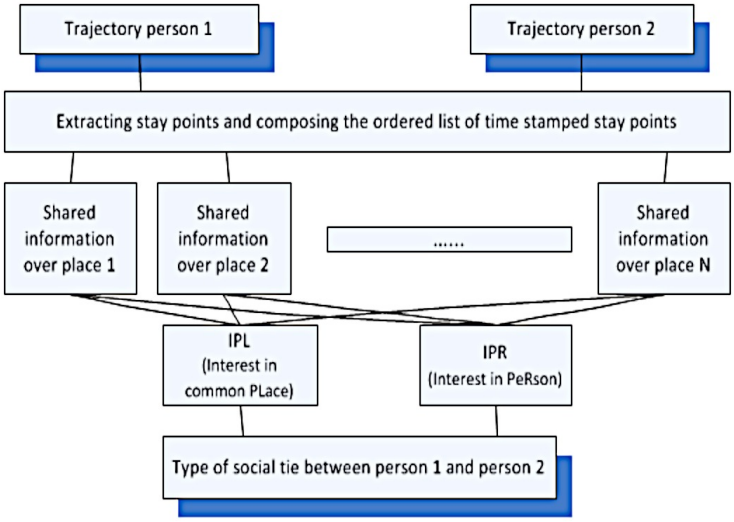


Figure 5-1 The process of extracting the type of social tie with our heuristic approach

Before we define these indicators, we explain two key observations related to the social behavior of people:

***Observation 1*:** The social tie between people may cause correlation in their visits *only to some* places and *not to all* the places that they visit. By correlation, we mean simultaneous absence and presence at a place. For example, ordinary friends may have correlation in visit to places such as cafés and restaurants (not at work) while people who work in the same place only have correlation in visiting their working place and not in visiting other places. When the visit of two people to their work place is correlated such that they are present at work on the same days, are absent on the same days, and work late on the same days, this is an indication that they may be socially related (e.g. they work on the same project). The fact that these persons visit different places when they are absent is not important anymore in deriving any conclusion about their social tie. Therefore, it is better to define the mutual information of people for each single place separately and to ignore the information content of correlation between two persons on the entire set of stay-points.

***Observation 2*:** People's intention of visiting a place is related to the social tie they have. Being with a friend is one of the primary reasons of visiting a place by those who have friendship tie. This means that two persons usually go to a café to be with each other and there is a friendship relation between them before going to the café. *The correlated visit of friends to different places are normally of low frequency and short duration*. Acquaintances, however, come to know other people as a *consequence* of their presence in a special place and not because they intend to be with those people. For example people start working and then get to know their colleagues and there is no acquaintance relationship between them before they start working. *The correlated visits of acquaintances to a specific place normally happen with high frequency*. Therefore, a solution to distinguish between different classes of social ties from visits to a set of places can be using measures that make a distinction between these two types of interests (in a person (infrequently visited places) or in a place (frequently visited places)) by considering the frequency of correlated visits.

Based on these two observations, we propose to compute mutual information content of mobile entities at each stay-point separately and then use the results in computing two indicators, which show the interest in person or interest in stay-points. Each indicator will accentuate the correlation between two mobile entities in visit to specific type of stay-point. One of these indicators will emphasize on correlation between two mobile entities in terms of their visit to frequently visited places and the other on infrequently visited places. In other words, the first indicator is an interest indicator for a stay-point (*IPL*), implying that being at the stay-point is the purpose of the visit, while the second one (*IPR*) is the interest indicator for a person, implying that being with a person is the reason behind the

visit. Using different combination of these two indicators we can discover people's social tie. We continue this section by providing a number of definitions, which are required to further define these two interest indicators.

***Definition 5.1**: Shared information content (SI) of two persons i and j for the time they spent at stay-point a is defined as:*

$$SI_a(i,j) = log\frac{p_a(i,j)}{p_a(i)p_a(j)} \tag{5.3}$$

Where $p_a(i)$ is the probability of a person i being at stay-point a, while $p_a(i,j)$ is the joint probability of two persons i and j being at stay-point a. One should note that this measure is different from the original mutual information. In contrast to mutual information, we only measure the information of simultaneous visits of two persons to the same stay-point using shared information content and not the combinations of stay-points at which one or both of these persons are absent.

***Definition 5.2:** Normalized shared information (NSI) content of two people i and j for the time they spent at stay-point a is defined as follows:*

$$NSI_a(i,j) = \frac{2 \times p_a(i,j) \times SI_a(i,j)}{H_a(i,j)} \tag{5.4}$$

Where $H_a(i,j)$ is computed as follows:

$$H_a(i,j) = log\big(p_a(i)\big) + log\big(p_a(j)\big) \tag{5.5}$$

We use $SI_a(i,j)$ and $NSI_a(i,j)$ to define two indicators for shared information created by the interest in (i) common stay-points and (ii) persons. Considering that different set of stay-points provide different information about social ties, each of these indicators accentuate the value of shared information content from the relative important set of stay-points (i.e., frequently visited stay-points and infrequently visited stay-points). The maximum value of each of these indicators will be 1.

***Definition 5.3:** The indicator of shared information due to **Interest in common Place (IPL)** for two persons i and j over a set of stay-points $A = \{a_1, \ldots, a_N\}$ with $(NSI_a(i,j) > Th)$, where Th is a predefined threshold is defined as:*

$$IPL = \sum_{a \in A} NSI_a(i,j) \tag{5.6}$$

As seen in Definition 5.3, for computing $NSI_a(i,j)$, the fraction of time that people spend together at each stay-point denoted by $p_a(i,j)$ is scaled by the information

they share at that stay-point $SI_a(i,j)$. This way, we put more focus on the shared information content over stay-points which are visited regularly and have higher $p_a(i,j)$. As mentioned before, regular visit is an indication of interest in a place. The IPL indicator value should be high for people who work, study, or live together. This indicator represents the information that two persons share over the whole observation time. The longer the amount of time that two persons spend together, the higher the effect of their shared information content on IPL.

Definition 5.4: *The Indicator of shared information due to* ***Interest in Person (IPR)*** *between two persons i and j over a set of stay-points* $A = \{a_1, \dots, a_N\}$ *with* $(NSI_a(i,j) > Th)$ *is defined as:*

$$IPR = \frac{p_{min}}{N \times SI_{max}} \sum_{a \in A} \frac{SI_a(i,j)}{p_a(i,j)} \tag{5.7}$$

In this equation, p_{min} is the lowest probability possible for a person over a stay-point $(1/t_{stop})$ where t_{stop} is the minimum stay time used to extract the stay-points, SI_{max} is $log(1/p_{min})$, and N is the total number of stay-points. We add the condition $(NSI_a(i,j) > Th)$ to prevent mistakenly ranking the low shared information content of two persons over a stay-point as being high because of the low probability of occurrence.

As opposed to IPL indicator, which scales the shared information content of two persons by the fraction of time they spend at that stay-point to focus on information shared over *frequently* visited stay-points, the IPR indicator divides the shared information content of two persons $SI_a(i,j)$ over the fraction of time they spend together $p_a(i,j)$. IPR accentuates the shared information content of two persons at stay-points visited *infrequently* (with lower $p_a(i,j)$). The shorter the amount of time two persons spend together, the higher the effect of their shared information content on *IPR* will be.

5.4.4 Identifying ties based on interest in location (IPL) and in person (IPR)

Different combinations of the two IPL and IPR indicators can show various types of social ties between people. The correlation between people in regular visits to their working place will be shown by their high IPL. These people may visit some random places together as well. For example, imagine a group of people who work in the same building. They mutually have high IPL. Among these people, those who work in the same group may spend some time in another place for a social activity. This will cause their IPR to slightly increase. This small amount of IPR will help in distinguishing the members of this group from all the rest who work in the same building and have lower probability of acquaintance. Cohabitees or buddies (those

who work or study together as well as perform non-frequent activities) might have both high *IPR* and *IPL*. The difference between these two groups is distinguishable if the time of day when activities due to interest in place are performed is also taken into account (for example, cohabitees will have high *IPL* during night-time and day-time, buddies will have high *IPL* only in day-time). Ordinary friends who do not work or study together may only visit each other once in a while and in some random places. Their correlation in such stay-points will cause their *IPR* measure to increase considerably.

We summarize combinations of these two indicators with respect to the type of social tie they represent in Table 5-4.

Link type	*IPL*	*IPR*
Acquaintances (with high probability)	High	Low
Acquaintances (with low probability)	High	Zero, Extremely low
Cohabitees	High (Night time)	High
Friends (buddies)	High (Day time)	High
Friends (ordinary)	Low	High
No relation	Zero, High	Zero-low

Table 5-4 Link types based on *IPL* and *IPR* indicators

The pseudocode of Algorithm 5.1 (to discover social ties, is presented below:

Algorithm 5.1 (LinkDescription)

INPUT: $\boldsymbol{D} = \{\boldsymbol{P_1}, \boldsymbol{P_2} \ldots \boldsymbol{P_N}\}$ **(data set of trajectories from people)**
OUTPUT: $\boldsymbol{L} = \{\boldsymbol{L_{1,2}} \ldots \boldsymbol{L_{N-1,N}}\}$ **(set of link types)**

For each $(P_i \in D)$
    Extract the stay-point and add them to the list *SPL*;
End for
For each $(P_i, P_j \in D)$
    For each $SP_k \in SPL$
        Measure $SI_{SP_k}(P_i, P_j)$;
    End for
    Measure *IPL* and *IPR* using $SI_{SP_k \in SPL}$;
    Set $L_{i,j}$ based on *IPL* and *IPR* ;//using Table 5-4
End for

Considering *Example* 1, after measuring *IPL* and *IPR* indicators using Algorithm 5-1, we will have the values presented in Table 5-5 and 5-6:

IPL	**Alice**	**Bob**	**Chuck**	**Linda**
Alice		0.01	0	0
Bob			0.23	0
Chuck				0.76
Linda				

Table 5-5 *IPL* indicator for Example 1

IPR	**Alice**	**Bob**	**Chuck**	**Linda**
Alice		0.14	0	0
Bob			0.003	0
Chuck				0.14
Linda				

Table 5-6 *IPR* indicator for Example 1

An important issue to be considered in interpreting these results is the role of time. As the observation time increases some of the above values change. As seen in Table 5-1, in this example for Linda and Chuck the *IPL* indicator measure is higher than that of Alice and Bob and also of that of Bob and Chuck. By extending the time of observation this indicator will decrease rapidly for Alice and Bob, while it stays the same for Chuck and Linda. The high value of *IPL* for Linda and Chuck is due to their high shared information content for the time they spend at home, while Alice and Bob have different working and living habits. *IPL* correlation of Bob and Chuck is 0.23 which is also a good indicator of their correlation at work. Their *IPL* will stay the same, as the observation duration increases. The indicator of interest in person for two couples (Alice-Bob and Chuck-Linda) is 37 times more than that of Bob and Chuck. If we extend the time of observation and Bob and Chuck keep working with each other and not visiting any random places together while they visit random places with their partners, then even this small interest between them will disappear. By increasing the time of observation, the correlation between the partners will stay the same. Using the combination of these two indicators based on Table 5-4, we can classify the social tie between these four people, results of which are presented in Table 5-7.

Link	Alice	Bob	Chuck	Linda
Alice		Friend (ordinary)	No-relation	No-relation
Bob			Acquainted (Low probability)	No-relation
Chuck				Cohabitee
Linda				

Table 5-7 Social ties of Example 1 based on *IPL* and *IPR* indicators

5.5 Evaluation

Evaluating the above-mentioned indicators in representing social ties is a challenging task. We require a mobility dataset, which has the necessary ground truth on the social tie information. We have chosen to use a dataset, which in addition to mobility data also contains some form of social context. MDC dataset [19, 20] collected by Nokia research is a newly released dataset collected by mobile phones with different data types such as phone call records and mobility data. Therefore, for our evaluations, in this section, we perform experiments with this dataset. In order to evaluate effectiveness of the above-mentioned indicators extracted from mobility data in representing social context, we compare the results suggested by these indicators with a number of features extracted from phone calls and short messages. Although our main aim is labeling the social ties with proper labels (mentioned in Section 5.3), using this dataset we can only prove that these indicators are powerfully representing the social ties inferred with the mobile phone related data.

Figure 5-2, 5-3, and 5-4 represent the social tie strength from message exchange and phone call information. The graphs shown in these figures are drawn based on the total number of seconds of phone call (in Figure 5-2), the number of messages exchanged (Figure 5-3), and the total number of attempts to contact (sent sms, received sms, outgoing call, received call, and missed call) (shown in Figure 5-4) between each two users. In these networks, nodes represent users and links are drawn based on the existence of contact between two users in terms of one of the previously mentioned factors. The link width is proportional to the phone call duration/sms counts/number of times reached between two users. As seen in these figures, evidently the social link strength in the graphs are versatile.

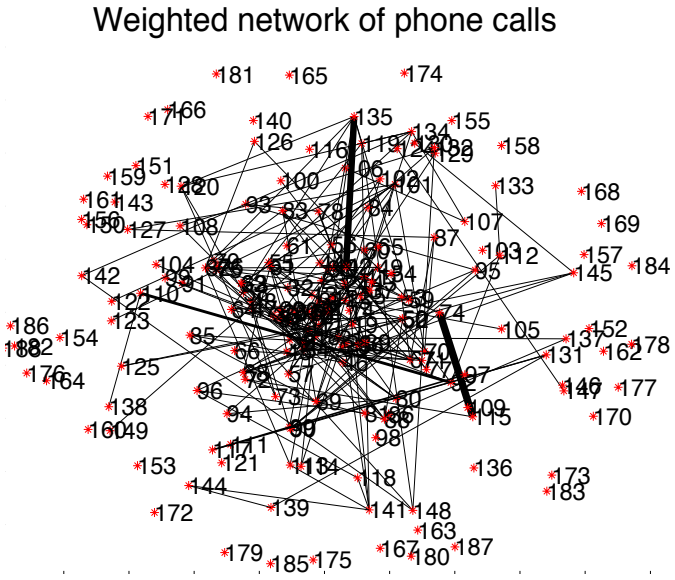


Figure 5-2 Wighted network of phone conversation duration

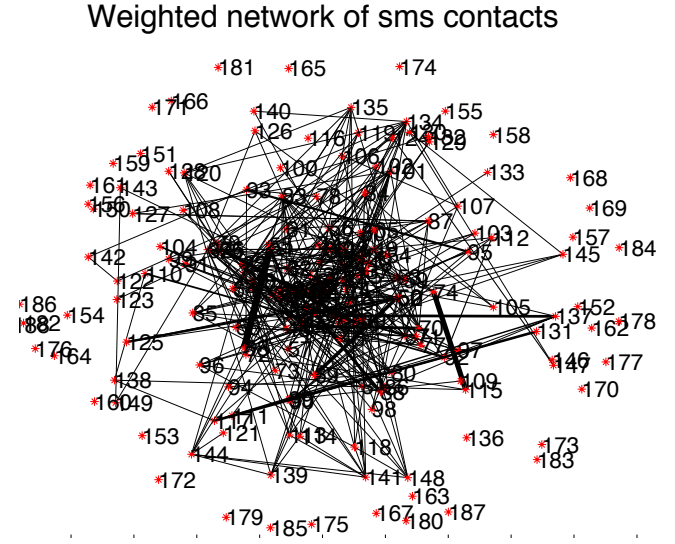


Figure 5-3 Wighted network of number of messages exchanged

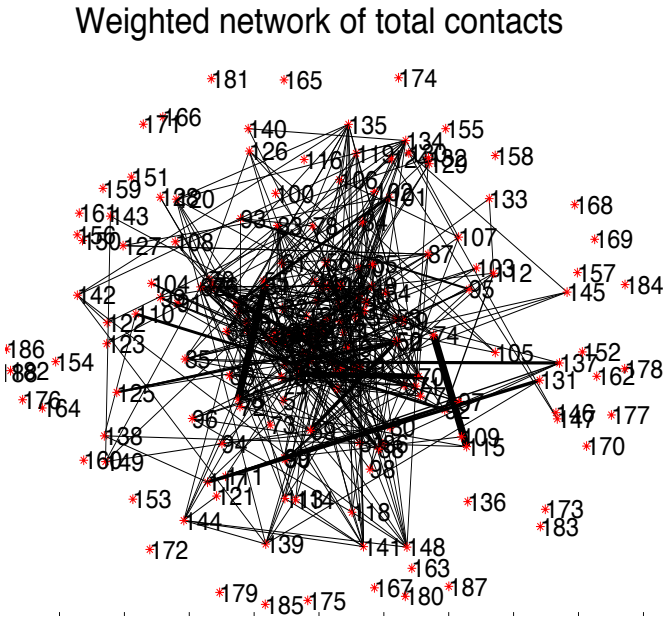


Figure 5-4 Wighted network of total attempts to contact

Assuming that phone call/sms features can strongly represent the social tie between people, in this section we follow two directions. Firstly, we investigate whether there is a meaningful correlation between the proposed mobility based social indicators and the phone call/sms based indicators. Secondly, we compare the results achieved using *IPL*/*IPR* indicators to the ones based on the total co-location indicator. The co-location indicator is the total amount of time two persons have co-existed at the same place. In what follows, we describe how to proceed with studying the relationship between mobility indicators (*IPL*/*IPR*/co-location) and the phone call/sms indicators.

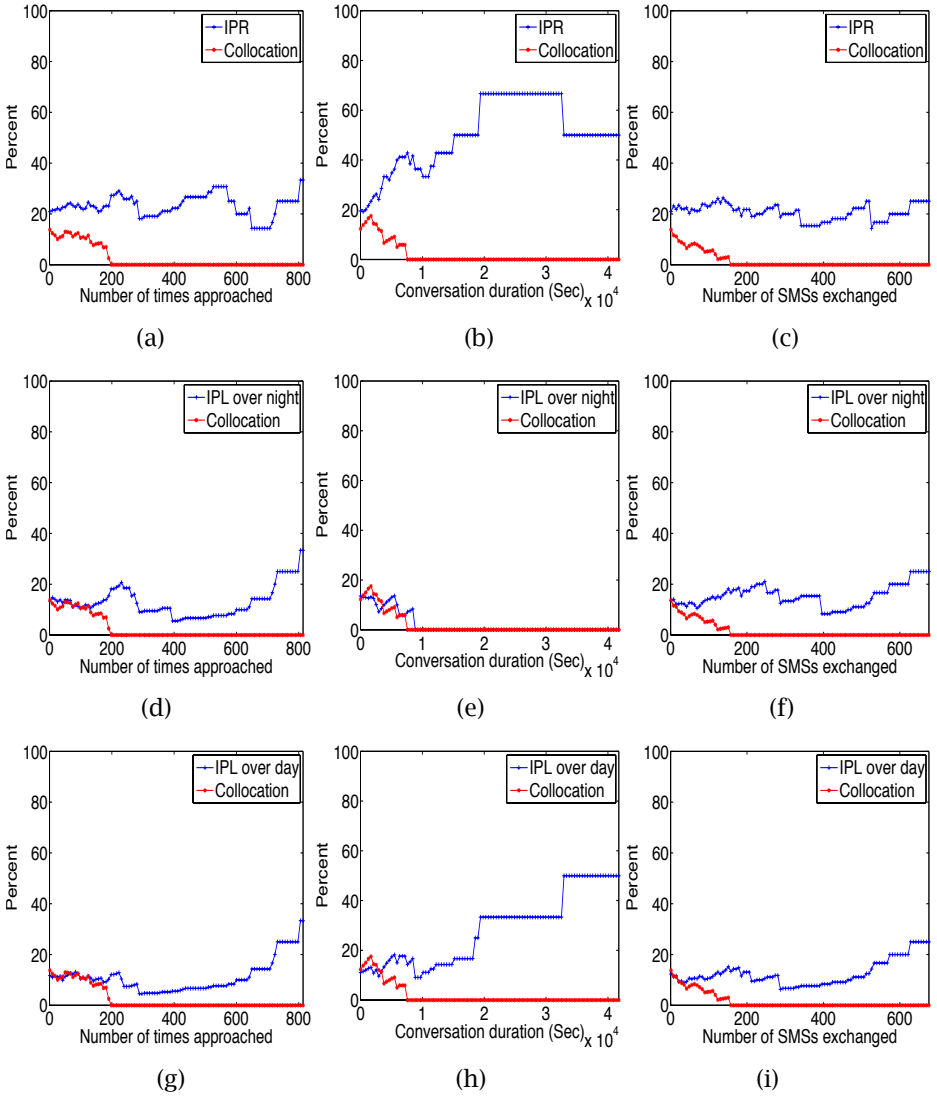


Figure 5-5 Comparing the *IPL*/*IPR* indicator and co-location indicator in representing the strength of phone call and sms features. Each graph shows the percentage of people with a strong mobility indicator in terms of the strength of tie extracted from phone call/sms features

Correlation coefficient [229] is a popular indicator for studying correlation between two random variables. However, this value can only represent linear relationships. Co-location indicators and phone call/sms indicators have different natures and their strength is not comparable. As mentioned before, phone call/sms indicators represent direct interaction while co-location indicators do not. In other words,

even a short phone call strongly represent interaction, while large amount of co-location may still not represent any interaction (e.g. working in different floors/department of a building). Therefore, a common correlation metric such as correlation coefficient does not represent the relationship we are looking for. In order to find meaningful relationships, we look into both strong and weak co-location indicators and see if there is a relationship between the strength of the phone call/sms indicators.

Figure 5-5, represents how strength of a mobility indicator is related to the phone call/sms indicators. In order to identify pairs with strong *IPL*/*IPR*/co-location indicators, we chose the ones, which are above one standard deviation over the mean of the indicator (as we are not interested in people with average *IPL*/*IPR* indicators). The graphs show the percentage of people with strong co-location indicators in terms of their phone/sms tie. Generally, if there is a positive relationship between the two groups of indicators (*IPL* / *IPR* /co-location-phone/sms), the percentage of people with strong *IPL*/*IPR*/co-location indicators should increase as the strength of phone call/sms indicators increase.

As seen in Figure 5-5, compared to the ordinary co-location indicator, *IPL* and *IPR* indicators can better represent the strength of social tie. This is due to the fact that for both of these indicators, percentage of people suggested by them increases more as the strength of the social tie represented in term of phone call/sms increases. As shown in Figure 5-5-b, especially the *IPR* indicator and the *IPL* indicator over day can successfully represent the social tie strength based on the phone call/sms features. The difference between these *IPL*/*IPR* indicators and the total co-location index is that the information content extracted from the visits from different places has been able to somehow make a differentiation between accidental co-locations and more correlated visits. Especially, the results achieved from the *IPR* indicator (Figure 5-5.a-c) are able to better represent social tie strength shown by call/sms as they focus on correlation of two people on visit to infrequently visited places (such as cafes, bars, and cinemas). All in all, we can conclude from Figure 5-5 that considering that the strength of phone call/sms features can relate to strength of social ties, *IPL* and *IPR* indicators can better represent social tie strength compared with the total co-location indicator.

Graphs presented in Figure 5-5 illustrate the relationship between strong mobility indicators and phone call/sms features. Figure 5-6 illustrates the relationship between weak mobility indicators and phone call/sms features. If there is a relationship between these two group of indicators, the percentage of people with weak indicators should decrease with the strength of the phone call/sms indicators. For weak indicators respectively, we choose cases under one standard deviation below the mean. We see that the percentage of people with weak co-location indicators and strong phone call/sms indicators is very low and it

decreases rapidly as the strength of the phone call/sms indicator increases. The *IPR* indicator is performing slightly better than the total co-location indicator as it decreases faster as the strength of phone call/sms feature increases. Generally, Figure 5.6 shows that all co-location indicators can represent the weakness of social tie.

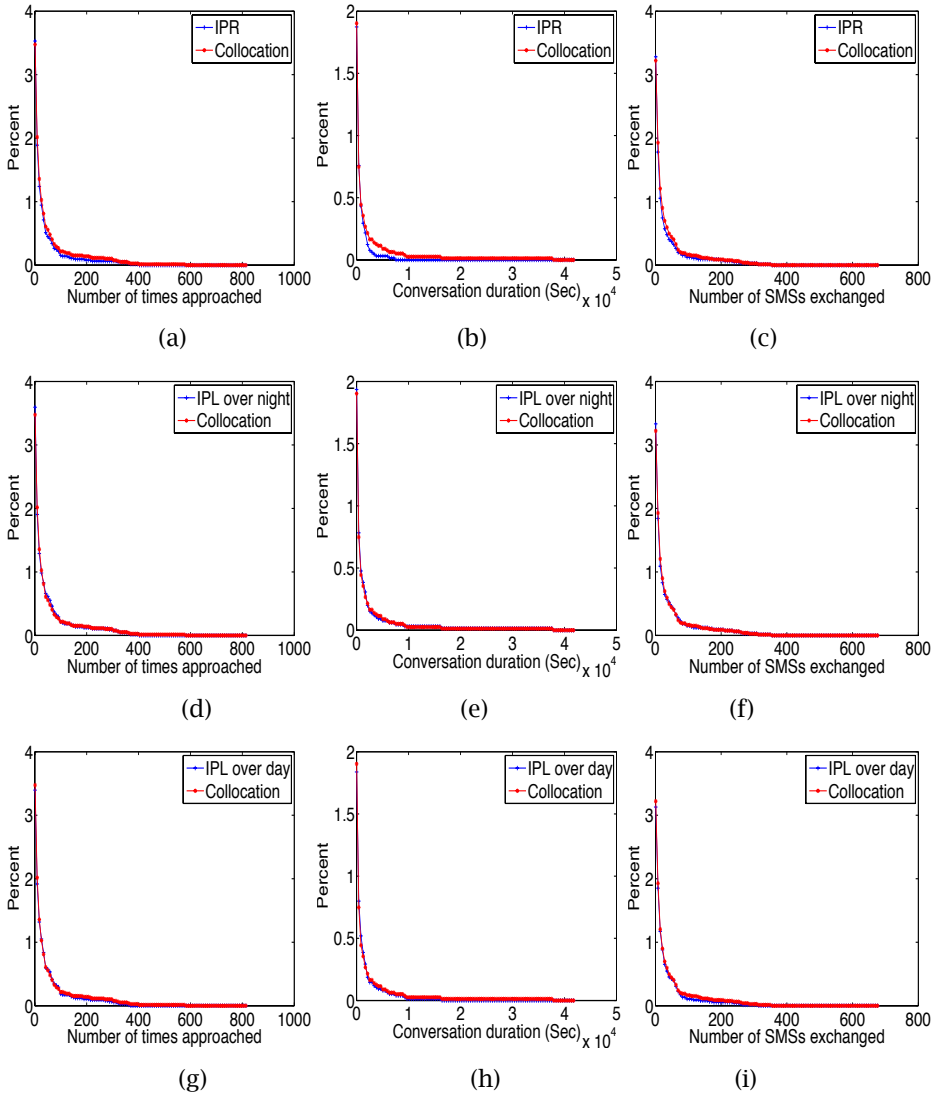


Figure 5-6 Comparing the *IPL*/*IPR* indicator and co-location indicator in representing the strength of phone call and sms features. Each graph shows the percentage of people with weak mobility indicators in terms of the strength of tie extracted from phone call/sms features

It should be mentioned that these graphs are for *IPL/IPR* indicators over zero. We observed that for *IPL/IPR* indicators equal to zero the percentage of people with zero phone call is about 100 percent. This large difference, made the graphs unreadable. Therefore, we did not represent them.

5.6 Case study

5.7.1 Case study using Dataset 1

In the first case study, we investigate the usage of indicators on Dataset 1 for a period of 21 days. The study group was composed of two couples (#1 & #2 and # 4 & #5) and two other colleagues (#3, and #6). All of these people work in the same building. Persons #1 and #2 mostly visit different places together. They only have very little difference in working hours because one works later. Persons #4 and #5 have very similar activities at work but normally one of them does some extra activities such as shopping alone. This couple has visited several random places together. Person #6 works one day less than the other five persons and lives in another city. The two couples once visited person #6 at his home. Person #5 is a visiting researcher who does not have any special social tie with the other five persons. He has only been at the same stay-point with person #1 and #2 accidently once in a super market.

We used the method proposed in [230] to extract the stay-points. Each stay-point is a set of spatial locations within the maximum radius of 100 meters of where people had stayed more than 1 hour. We later merged the stay-points closer than 100 meters. Due to high density of places, sometimes one stay-point does not necessarily show one specific attraction but a group of them (a shopping center rather than a single shop). We extracted 23 places as stay-points, 11 of which were at least visited by 2 persons.

Figure 5-7 compares visits of the five persons in terms of the time they have stayed in the 11 extracted stay-points. We do not represent the visit to other 12 stay-points as they do not represent the social tie (being visited by only one person). Stay-points 1, 5 and 8 are the houses of the two couples and person #6, respectively. Stay-point 2 is the place in which these five persons work. Stay-point 4 is an area in the city center with shopping centers, Stay-point 10 is a gym and the rest of the stay-points are random places in which at least 2 people had stayed.

As seen in Figure 5-7, the distinction between low and highly visited stay-points is evident. The only stay-point that all candidates have visited is their working place (Stay-point 2) and their visit to this stay-point has been relatively high but less than their houses.

After measuring the shared information content of each pair based on Definition 5.1 over all stay-points, we use the results to measure *IPR* and *IPL* indicators for each pair. Figure 5-8 and Figure 5-9 illustrate the obtained results. Since the results are symmetric, we only show the values over the diagonal line in Figure 5-8 and Figure 5-9. Furthermore, considering the fact that the use of these indicators is meaningless for comparing one person to himself, we also omit the values on the diagonal line for better visibility.

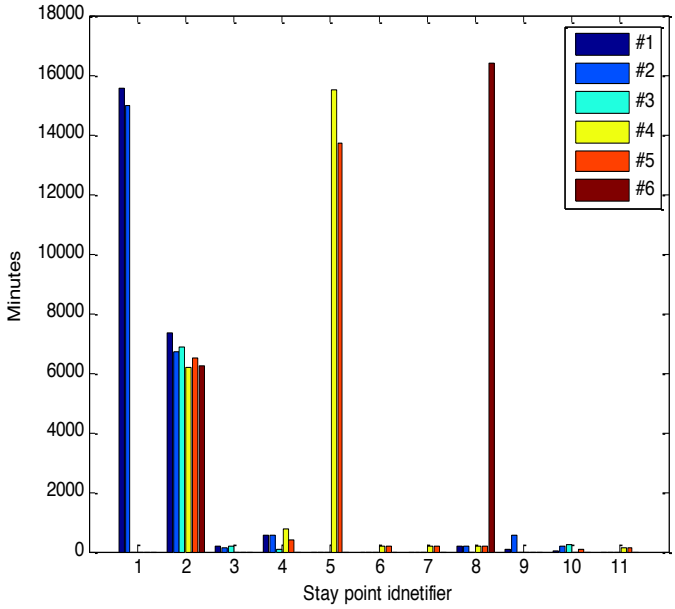


Figure 5-7 The amount of time spent in each stay-point by different candidates

The *IPR* indicator shows the information that two people share relative to the time they spend together. In this case, the effect of information that people share in frequently visited stay-points such as Stay-point 1,2 and 5 will be degraded. By looking at *IPR* indicator measure in Figure 5-9, we realize that the level of *IPR* is high for the two couples than the rest. This high *IPR* value is due to the correlated visit to infrequently visited stay-points such as Stay-points 3, 4 and 6-11. The couple (#4&#5) have visited more random stay-points compared with the first couple and naturally their *IPR* is higher. The level of interest between person #2 and person #5 is also high due to their coordinated random visits to a gym and their visit to the house of person #6.

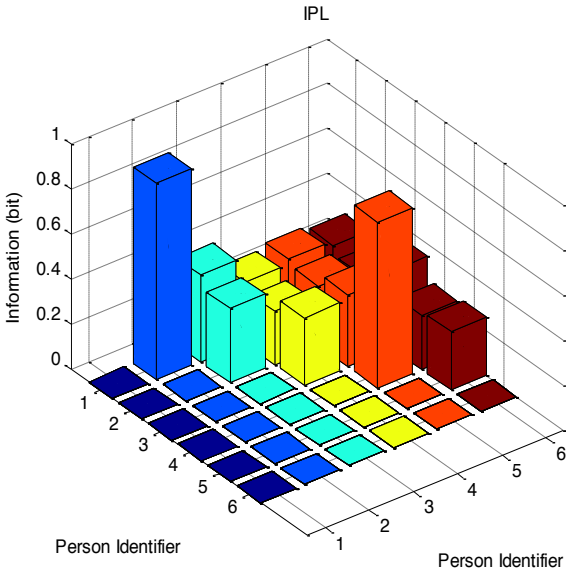


Figure 5-8 Information shared due to Interest in place (*IPL*)

As seen in Figure 5-8, the *IPL* value of all these persons is higher than 0.2 bits. This is due to the considerable amount of time they have spent in Stay-point 2. Moreover, the *IPL* value is also high for the pair (#1&#2) and pair (#4&#5) due to the long time they have spent together living in the same place (at Stay-point 1 and 5, respectively).

An interesting observation is that, the time that two couples have spent with person #6 at his house (at Stay-point 8) has only brought information on their social ties with each other while as seen in Figure 5-9, the *IPR* value of person #6 with person #1, person #2, person #4, and person #5 is still low. Looking at Figure 5-7, it is seen that the probability of person #6 being at his house (at Stay-point 8) is considerably high so this is not a good indicator of his interest in seeing the two couples. This also seems logical as it was also possible that the two couples were in somebody else's house who lived in the same building where person #6 lives. However, if person #6 had spent more time with any of these four persons in a random Stay-point then, the shared information was more helpful in identifying their social tie.

Another interesting observation is that although person #3 has spent some time at Stay-points 3, 4, and 10 which other persons have also visited, the value of second indicator does not show distinguishable *IPR* between this person and the others. The reason is that the visit of person #3 has either happened at other times or the

result of correlation was under the threshold used by Definition 5.2 which could be considered as an accidental co-occurrence.

The results illustrated in Figure 5-8 and Figure 5-9 show that a clear distinction can be made between the colleagues who only work at the at same stay-point with no specific social tie with those who have social ties outside work.

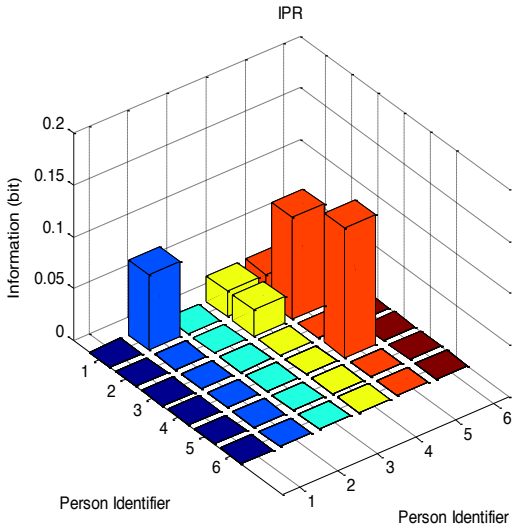


Figure 5-9 Information shared due to Interest in person (*IPR*)

5.7.2 Case study using Dataset 3

In the previous case study, we showed how *IPL* and *IPR* indicators can provide information about the social ties between people. In this section, we present the results of calculating *IPL* and *IPR* indicators for the Capricorn dataset. Although the social tie between animals might be different from that of humans (e.g. friends and work-related ties), the information being extracted by the above-mentioned indicators can be useful in identifying the patterns of co-location. As shown before in Figure 4-9, there is generally one stay-point in Dataset 2 where two capricorns have co-existed. Figure 5-10 shows the amount of hours that these animals have co-existed. As seen, two capricorns have spent a considerable amount of time at the first stay-point. From the *IPL* and *IPR* indicators, illustrated in Figure 5-11 and Figure 5-12, it can be seen that Capricorn #1 and #2 have both high *IPL* and *IPR* indicators. This is due to their more similar daily mobility habits. An *IPL* indicator with more than 0.1 bits represents a special form of social tie between two animals, which has resulted into high correlated behavioral mobility pattern.

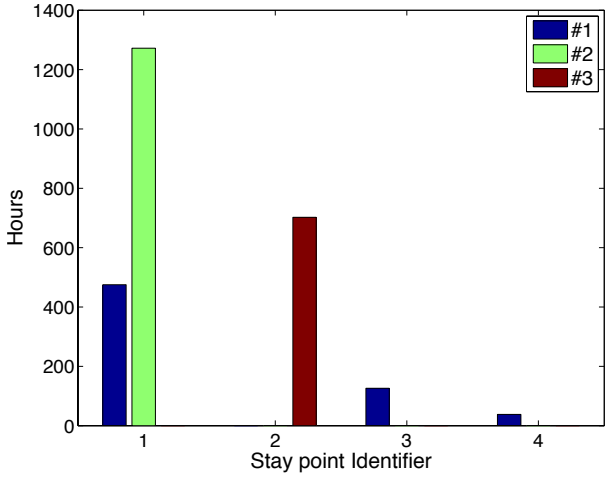


Figure 5-10 The amount of time spent in different stay-points by the three capricorns

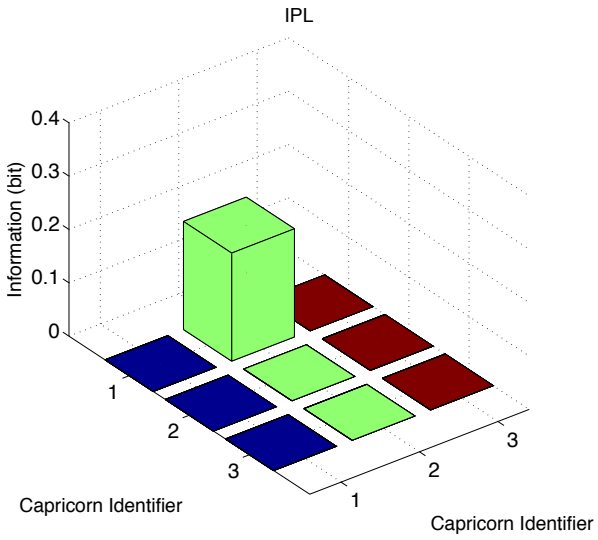


Figure 5-11 Information shared due to Interest in place (*IPL*)

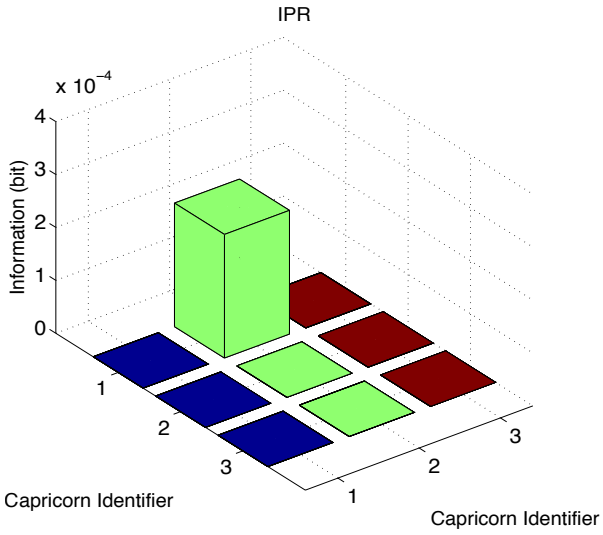


Figure 5-12 Information shared due to Interest in person (*IPR*)

5.7 Summary

In this chapter, we presented a method for differentiating between different types of social ties between mobile entities. We defined shared information content metric based on mutual information to extract the information content which shows correlation between two mobile entities at a certain location. Next, we used the information content from each place in computing two indicators. These two indicators represent the interest in common places (*IPL*) or in person (*IPR*). We further used these indicators to identify the type of social tie between two mobile entities.

We have shown that, the proposed indicators are useful in identifying the existence of a social tie between two mobile entities as well as in describing the type of social tie using mobility data only. A potential application of these indicators can be used as an additional tool in improving the performance of online location-based social networks.

Our evaluation results using Nokia Mobile Data Challenge dataset showed the superiority of IPL and IPR indicators compared to normal co-location indicator in representing the social tie information derived from the phone call/sms data. In our case study, we showed how these indicators show the specific form of correlation both between a group of people and capricorns. One interesting future research topic is studying extraction of social ties with a dataset with more ground truth on social tie type.

Last but not least, such form of social context analysis should always be performed considering privacy requirements. Although, in this chapter we did not address the privacy requirements, research is currently being performed in this direction. For instance, in [231] a framework is proposed to provide multi-layer privacy requirements to safe-guard users private information before the data is analyzed for any purpose.

5.8 Acknowledgement

Portions of the research in this chapter used the MDC database made available by Idiap Research Institute, Switzerland and owned by Nokia.

Chapter 6

Trajectory Compression

Human beings and animals exhibit common patterns in their activities and movements. Transmission and collection of all the repetitive data representing these activities and movements is neither needed, nor is it efficient (in terms of processing, memory and communication overhead). Mobility data can be summarized and expressed compactly with respect to the patterns it conforms to. In this chapter, we propose two solutions to efficiently collect and store spatio-temporal mobility data. These two techniques reduce the communication, memory overhead, and consequently energy consumption required for collection and storage of mobility data. In particular, we first propose a model based compression technique to predict the future samples continuously and to send data when it is needed. We then propose a technique, which further reduces the energy consumption of the data acquisition devices by adaptively changing the sampling rate with respect to the predicted movement of the mobile entity. We evaluate the performance of the above-mentioned techniques by comparing them with commonly used trajectory compression techniques in terms of the memory saved, total error in trajectory representation, and energy consumption (both in sensing and communication).

6.1 Introduction

The number of mobile entities equipped with some form of location-acquisition device is increasing rapidly. In addition to the challenges enforced by data analysis techniques to interpret and understand mobility data, management of such huge amount of mobility data has also introduced new challenges. The mobility samples collected by sensing devices are profoundly redundant. On the one hand, due to limited speed and high spatio-temporal resolution of data collection, consecutive visited locations may be similar (i.e., existence of spatial correlation). On the other hand, as shown in [232], there is a great number of movement patterns, which in the longer term are repeated with different frequencies (spatio-temporal correlations). While this abundance of data produces different capabilities and possibilities for mining habits and behaviors, the massive volume of spatio-temporal data imposes unnecessary burdens on the data acquisition, transmission, and storage systems, if all data should be transmitted and stored. More specifically, such volume of mobility data increases the system costs in 4 different ways. Firstly, there are costs in terms of storage capacities of current devices. As shown in [233],

assuming that GPS data is acquired over 10 second intervals, without applying any data compression, 1 Gigabyte of storage capacity is required to store the data of just 4000 objects for a single day. Secondly, the cost of sending such data in terms of price is high ($5 to $7 per Megabyte) [234]. The third cost is the processing cost imposed by different pattern recognition algorithms. The complexity of such algorithms is always proportional to data size. Last but not least, the energy cost of sensing and acquiring location data from GPS satellites is considerably high. As shown in [235], the amount of energy spent during a single GPS sample acquisition can be as high as 60 times more than the amount used when sending it over the radio of a wireless sensor node. High-energy consumption is the major performance bottleneck of networks, which acquire GPS samples. It drastically reduces the lifetime of a system, which is intended, for collecting mobility data from GPS satellites in comparison to the other types of data.

6.1.1 Contributions

If mobility data generated by GPS enabled devices is efficiently compressed, lifetime of the mobility acquisition device will be improved and the memory will be efficiently used. Respectively, our contributions in this chapter are two fold and can be summarized as:

- Proposing a technique to use the patterns in the mobility data to represent trajectories in a compressed way.
- Extending the abovementioned approach by an adaptive sampling technique to increase the lifetime of the mobility-sensing device.
- Comparing the proposed solution with existing solutions in terms of total error, memory saved, and (sensing and communication) energy consumption using real and synthetic datasets.

The rest of the chapter is organized as follows. Section 6.2 presents the related work. Formal definition of our problem statement is described in Section 6.3. The detailed description of our approaches are provided in Section 6.4. Evaluation results and case studies are reported in Section 6.5 and 6.6, while Section 6.7 is the summary.

6.2 Related works

Various data compression algorithms have been designed in literature for different purposes. In general, these algorithms either work on a lossy or loss-less basis. In lossy compression, redundancies in data (or information) are subject to permanent elimination. Lossless compression, on the other hand, compresses data by identifying and reducing statistical redundancies. This implies that the original

data can be later reconstructed with no further loss. Trajectory compression is normally performed through lossy compression. The reason is that [233] (i) raw mobility data contain a substantial amount of noise which is of no use to be recovered (ii) understanding mobility behavior is achieved easier when smaller amount of data with behavior-related context is available. The existing solutions for trajectory compression can be classified into two general categories, i.e., geometric compression techniques and non-geometric compression techniques:

Geometric compression algorithms: Naturally, the very first methods proposed for trajectory compression were based on geometrical trajectory simplification. Line and curve simplification have been used for compression and de-noisification of trajectories [81, 236, 237]. This simplification is based on a maximum distance error function. Such algorithms are concerned with approximating the trajectory with another one within a predefined error tolerance. The first and commonly used trajectory compression technique on this basis is Douglas-Peuker [236]. As this algorithm is purely spatial, it does not capture the temporal characteristics of the trajectory. To increase the efficiency of Douglas-Peuker and make it better suited for spatio-temporal trajectories, TD-TR [233] was proposed based on the concept of synchronous Euclidean distance. Synchronous Euclidean distance uses the information achieved from speed of the trajectory for synchronizing and identifying the position of measurements to be discarded. Bellman's algorithm [204] is also another well-known technique which uses dynamic programming to minimize the area between the original trajectory and its compressed representative.

The above-mentioned algorithms were originally meant to be applied on data in a batch mode. They were later modified through window-based alternatives where a sliding or an opening window was used to make the previous algorithms suited for streaming data [233, 238]. Following the same goal, a bottom-up algorithm was proposed in [239]. This technique starts from the consecutive sample points and approximates a new trajectory by merging the consecutive segments into one line segment with the least error. Dead reckoning [159] and priority queues [240] are also proposed for provision of the use of previous algorithms in streaming setting.

Following the same research track on geometric compression, recently a number of researchers have considered optimization through approximating the trajectory by a small "coreset" of data [241-243]. This coreset is a small set which approximately represents the original data. For instance, in [242] points in the trajectory are approximated by k number of cylinders with a certain radius that cover the trajectory. By applying map-reduce techniques, authors of [241] proposed a method to reduce the size of the trajectory according to available memory. These methods are concerned with optimizing the error function.

Non-geometric compression algorithms: While looking at a trajectory from a geometric point of view is the first option for compression, there are also other redundancies in trajectories, which can be used for compression. Very recently, some researchers have looked into semantic trajectory compression [244]. The semantic features of trajectories such as stay-points and transitions between them have been used in order to compress trajectories. Authors of [245] utilize heuristic prediction to decide on the locations which should be preserved while compressing trajectories. This group of trajectory compression algorithms, referred to as Map matching algorithms, [246-248] map trajectories on available indexed maps from road networks and reduce the number of points required for representing trajectories. The downside to these techniques is their reliance on map data, which make them only applicable for cars, which have strictly structured movement. Recently, authors of [238] have proposed using compressed sensing in trajectory compression. In this technique a projection matrix is used based on a previously collected dataset from the same area.

In this chapter, we have a new view on trajectory compression. As shown in Section 4.3 of Chapter 4, by applying state-space model on trajectories we were able to find high-level patterns in the mobility data. Such information can also be used for efficiently compressing trajectories and even increasing the energy efficiency. To the best of our knowledge, such view on trajectory compression has not been taken into account before. The trajectory compression methods we propose in this chapter are based on reducing such form of redundancies in trajectories, which are represented in form of patterns. Generally, compared with the geometric compression algorithms, this trajectory compression goes beyond redundancies, which are captured in form of lines. It is light and suitable for compression in streaming setting. Compared with the non-geometric techniques proposed in the geometric setting, it does not require background information from maps and works also for entities with unstructured movement (humans and animals).

6.3 Problem Definition

As shown in Chapter 4.3, the complete movement track of a mobile entity can be modeled as transitions between states with their specific duration distribution. These states can explain behavior of the mobile object from observations in form of mobility data. For instance, these states can represent stays of the mobile object in specific places with certain duration distribution. Using the parameter-learning algorithm HHSMM (Algorithm 4.3), parameters of this model, i.e., $\lambda = (Q, O, A, B, C, \pi)$, can be learnt. These parameters are state transition probability matrix A, emission probability matrix B, state duration probabilities C, and initial state probability matrix π. Assuming that such parameters are learnt from the data collected for a specific initial duration (dependent on the duration of states found), the problem

we face in this chapter is to use such model for the purpose of compressive representation of trajectories and increasing the lifetime of the mobility data acquisition device.

6.4 Methodology

6.4.1 Assumptions

Before we continue with our proposed compression algorithms, in this section, we provide background information on data collection setting. We have a sensor network composed of at least two types of nodes. The base station, denoted by BS, is a powerful node capable of processing data and learning the parameters of the model. The sensor node S_i, which samples mobility data, is a mobile sensor node equipped with GPS carried by a mobile entity. In regular timestamps, each sensor node S_i sends the location it has acquired to the base station BS. The base station computes the parameters of the model $\lambda_i = (Q, O, A, B, C, \pi)$ and sends the corresponding parameters λ_i to each mobile node. Assuming that the movement model does not change during the observation period, both node types (BS, S_i) move on with prediction based on the exchanged model.

6.4.2 Increasing mobile node lifetime using a trajectory model

Considering that the model can be extracted in the base station using Algorithm 4.3 and be sent to the mobile entities, there are two possibilities to improve the lifetime of the mobile sensor node:

- ***Model-based trajectory compression:*** In which the mobile sensor node uses the model λ_i for *predicting the future locations* of the mobile entity and only sends those results, which are *not predictable* with the model. This way, unnecessary samples are omitted and the sensed trajectory is compressed before being sent to the base station.
- ***Model-based adaptive sampling*:** In which the mobile sensor node uses the derived model for changing the sampling rate so that, less samples are acquired when the mobile entity is stationary and more samples are acquired when the mobile entity is moving.

In what follows, we explain both of the above-mentioned techniques:

6.4.3 Model-based trajectory compression

In this section, we propose our technique for compressing trajectories using the pre-computed model (λ_i). In this model-based trajectory compression, both the base station (BS) and mobile nodes (S_i) use the model (λ_i) for synchronous location prediction. Sensor node carried by the mobile entity acquires the location of the entity at time (o_t). Concurrently, it makes prediction about the next position of the mobile object (o'_t) based on the model (λ_i) received from the base station and previous location measures ($o_{1\ldots t}$). In case the predicted location (o'_t) and the newly measured position o_t are within an acceptable error bound (provided based on the application requirements), the node does not transmit the position measurement o_t but will transmit it otherwise. When no measurement is received from the mobile node, the base station will predict the position at time t (o'_t) using its prediction model.

Given the parameters of the model (λ_i), the future state (q_t) and its duration (d) can be estimated using the forward algorithm [249]. When the parameters of the hidden Markov model is known, the forward algorithm can be used to calculate a 'belief state' which is the probability of a state at a certain time, given the history of evidence.

Assuming that τ_t denotes the remaining time of the current state q_t, then the forward variable $\alpha_t(m,d)$ that is the probability of the system being at state s_m, with remaining time d at time t is calculated by:

$$\alpha_t(m,d) = Pr\,[o_1^t, (q_t,\tau_t) = (s_m,d)] \tag{6.1}$$

We assume that we get mobility samples in each pre-specified time stamp. However, it is possible that not all observations are acquirable (due to cloud cover, or device mal-functions). This will lead to missing samples. As seen in Eq. (6.2), in case the observation is not valid ($t \notin T$), equal probability is considered for all states.

$$\alpha_t(m,d) = \begin{cases} \alpha_{t-1}(m,d+1)b_m(O_t), t \in T \\ \alpha_{t-2}(m,d+1), \qquad t \notin T \end{cases} \tag{6.2}$$

Where the initial condition is measured by:

$$\alpha_1(m,d) = \pi_m b_m(o_1) p_m(d) \tag{6.3}$$

Originally, the computational cost of the forward algorithm increases with the length of observations, as it uses all the measurements. Conforming to the Markov property, [250] future state only depends on the current state. Therefore, we only keep the memory of the last state, and its duration discarding the probabilities of the corresponding state. Thereby, the initial state can be calculated using the

corresponding line from the emission matrix. This process is explained in Algorithm 6.1.

Algorithm 6.1 (ModelBasedCompression)

INPUT: A (State transition matrix), B (Emission matrix), C (State duration matrix), PAI (initial state matrix), rate (Compression rate), preTimestamp, curTimestamp, repO (observation representing each state), pState (Previous state)
OUTPUT: nextTimestamp (next timestamp to be saved), preTimestamp, PAI

1: [$alpha$]= forward (A, B, C, PAI, O ($preTimestamp, curTimestamp - 1$));
2: [$state, remain$]=find ($state, remain$) **where**
3: alpha($state, remain$)==maximum(alpha);
4: **If** $state \neq pState$
5: Update $PAI, preTimestamp$;
6: **End if**
7: $predictedObs = repO(state)$
8: **If** distance ($predictedObs$, $O(curTimestamp)$)> th
9: Send ($O(curTimestamp)$);
10: update PAI;
11: **End if**

6.4.4 Model-based adaptive sampling

As seen above, having a trajectory model offers an opportunity for compressing trajectories through its predictive capability. When used as a basis for adaptive sampling, this capability can also help in increasing the mobile object's lifetime. Location acquisition consumes major part of the mobile nodes' energy resources. Therefore, cutting down on the number of samples, greatly reduces the node's energy consumption. Using hidden semi-Markov model as mentioned before in Section 4.4 each state will be assigned with a duration distribution. Accordingly, it seems logical to use the estimated duration of each state to adjust the sampling rate.

For those states with relatively longer durations, it is only necessary that the start and end of that state is sampled. At the same time, even when the mobile entity has a very repetitive behavior, the start and end times of states do not always occur at a fixed moment. Nonetheless, start and end times of states are of high importance as they represent changes in the trajectories. In order to avoid losing the unpredictable movements, which change the duration of the current state, we propose sampling less frequently during the state and more frequently near the end of the predicted state. This process is explained Algorithm 6.2 (ModelBasedSampling).

Algorithm 6.2 (ModelBasedSampling)

INPUT: A (State transition matrix), B (Emission matrix), C (State duration matrix), PAI (initial state matrix), rate, preTimestamp, curTimestamp
OUTPUT: nextTimestamp (next timestamp to be sensed), preTimestamp, PAI

1: $Curstate = findstate(O(curTimestamp))$
2: **If** $Curstate != Previousstate$
3: $PAI = B(:, O(preTimestamp))$;
4: **End if**
5: $[alpha]$= forward $(A, B, PAI, O(preTimestamp, curTimestamp - 1))$
6: $[state, remain]$ = find(state, remain) where alpha(state,
7: remain)==maximum(alpha);
8: **If** $remain > dur/rate$
9: $nextTimestamp = (curTimestamp + dur/rate)$;
10: $preTimestamp = curTimestam$;
11: **Else**
12: $nextTimestamp = (curTimestamp + 1)$;
13: **End if**

6.5 Evaluation

6.5.1 Complexity analysis

The complexity of Algorithm 6.1 is dependent on calculating the forward variable, finding the current state, and its remaining duration. The second task is composed of finding a state with the maximum probability in the transition matrix. This task is of $O(M)$ complexity where M is the number of states. Complexity of calculating the forward variable at time T from the start of the sampling is $O(MTD)$, where D is the maximum state duration. However, in case the forward variable is computed from the beginning of the previous state, as explained in Section 6.4.3, the cost of computing it will be reduced to $O(MD)$.

Complexity of Algorithm 6.2 is dependent on estimating the current state, calculating the forward variable, estimating the future state and its duration. Estimating the current state is performed through finding the state with the maximum probability in the emission matrix. The computational complexity of this task is of $O(M)$. Calculating the forward variable and estimating the future state and its duration is the same as Algorithm 6.1. Hence, the overall complexity of both of the Algorithms 6.1 and 6.2 is $O(MD)$.

The major difference between these algorithms, in terms of energy consumption, is in their cost of sensing. In the first algorithm, sensing is performed on a fixed

timestamp basis, in the second algorithm; however, sensing is performed based on the state duration.

6.5.2 Benchmarking

We compare the performance of the two algorithms suggested above with the following algorithms in the field of trajectory compression:

Uniform sampling [240]: Uniform sampling is basically down-sampling the trajectory with respect to the resources available. In this technique, choosing the time-stamps to sample the trajectory is based on fixed predefined intervals. The only advantage of this method is its simple implementation. On the contrary, due to the oversimplified choice of samples often critical points of trajectories are lost.

Douglas-Peuker [236]: This algorithm is one of the well-known algorithms in trajectory compression. It recursively partitions the trajectory and removes the furthest point from the resulted segment. When recursion is over, a new output curve is generated. An input error threshold $\varepsilon > 0$ provided as input to the algorithm guarantees that the Euclidean distance of the returned curve from the original does not exceed ε. This is the main advantage of this compression technique compared to the other lossy data compression techniques such as wavelets [251]. Another advantage of this technique is its simple implementation. Furthermore, it is proven that this algorithm achieves near-optimal saving at a far superior performance in terms of spatial error.

TD-TR (Top down-Time ratio) [233]: The distance metric used in Douglas-Peuker algorithm is the Euclidian distance, which is based on the perpendicular distance between a point and a line. This way, however, the temporal data inferred from the speed of the mobile entity is not taken into account. In order to take advantage of speed information, usage of Synchronous Euclidean Distance (SED) is proposed in [245]. For computing SED the difference between a point and its spatio-temporal image is considered. Given three points A, B, and C ($t_A < t_B < t_C$) the SED between point B and its estimation B' is calculated as:

$$SED(A,B,C) = \sqrt{(x'_B - x_B)^2 + (y'_B - y_B)^2} \tag{6.4}$$

Where

$$x'_B = x_A + \frac{x_c - x_A}{t_c - t_A} \times (t_B - t_A) \text{ and } y'_B = y_A + \frac{y_c - y_A}{t_c - t_A} \times (t_B - t_A) \tag{6.5}$$

There are different algorithms in literature, which can be used for comparison. We have chosen the ones mentioned above for specific reasons. The Douglas-Peuker algorithm acquires optimal result in terms of the spatial error in trajectory representation. TD-TR is optimal in terms of the spatio-temporal error. We compare the proposed algorithms with these two techniques to compare performance in

terms of error. As the model based adaptive sampling technique, also changes the sampling rate, we compare our proposed algorithms with the uniform sampling to see the amount of performance improvement in terms of error and energy consumption. We have chosen this technique since, to the best of our knowledge, there is no other previous work on changing the sampling frequency of mobile sensor nodes.

6.5.3 General features

There are certain generic features, which need to be taken into account in order to compare trajectory compression algorithms. These attributes are (summarized in Table 6-1):

Memory and Computational complexity: These parameters define the amount of memory and computational resources required for the algorithm to operate on trajectories. Efficiency of algorithms in terms of these parameters is one of the essential requirements of wireless sensor networks. As seen in Table 6-1, both Douglas-Peuker and TD-TR require considerably higher amount of memory and computational resources which is in the order of the size of the mobility data stream, while using the model based techniques, the complexity will always stay bound to limited number of states.

Mode of operation: This parameter defines whether the algorithm is executed on each newly measured sample or on the whole dataset. In case of the latter, the algorithm's mode of operation is batch. As seen in Table 6-1 the major problem of Douglas-Peuker and its synchronous version is their batch mode of operation.

Error bound adjustability: This parameter defines if it is possible to set the maximum distance of each point on the uncompressed trajectory and its compressed version. As explained before, the error bound of Douglas-Peuker and its synchronous version is fixed. This means that it is ensured that the trajectory data points are saved so long as they are further away from their estimation on the compressed version of the trajectory. For these algorithms, the error bound can be provided as input. In model-based compression, the data needs to be gridded (discretized with a grid) beforehand. Therefore, the error bound is dependent on the area of each grid cell. For model-based sampling this error bound is not assured.

Compression ratio: This parameter defines if it is possible to set compression ratio as input to the algorithm. Having this possibility is desirable as it helps managing and predicting the energy requirements and the memory resources. As seen, among the five algorithms compared in this section, only Douglas-Peuker and TD-TR do not support this characteristic inherently.

Method	Computational Complexity	Memory	Mode of operation	Error bound	Compression ratio
Douglas-Peuker	$O(nlogn)$	$O(n)$	Batch	Fixed	-
TD-TR	$O(nlogn)$	$O(n)$	Batch	Fixed	-
Uniform sampling	$O(1)$	-	Online	-	Adjustable
Model based[9]	$O(MD)$	$O(M^2 + G + MD)$	Online	Fixed over> grid cell	Adjustable
Model based sampling	$O(MD)$	$O(M^2 + G + MD)$	Online	-	Adjustable

Table 6-1 Compariosn of generic attributes of different compression techniques

6.6 Case study

We use Dataset 1 and 2 for comparing compression algorithms. As these algorithms work on different bases and different input requirements, we repeated compression for each algorithm by changing the input parameters separately. We use the total error, maximum error, and memory footprint reduction as performance metrics for our comparisons. The total error and maximum error, as defined in Eq. (6.6) and Eq. (6.7) are used to compare the average performance and extreme deviations, respectively.

- **Total error metric:** This metric represents the total sum of distance between the trajectory and its compressed version. In order to calculate this distance, we use the distance between points (x_{ti}, y_{ti}) and their compressed representative $(x_{ti}^{'}, y_{ti}^{'})$.

$$TotalErr = \sum_{i=1}^{n} \sqrt{(x_{ti} - x'_{ti})^2 + (y_{ti} - y'_{ti})^2} \tag{6.6}$$

- **Maximum error:** This metric represents the maximum difference between points on a trajectory and their compressed representative.

$$MaxErr = Max\{(x_{ti} - x'_{ti})^2 + (y_{ti} - y'_{ti})^2\}_{i=1}^{N} \tag{6.7}$$

- **Memory saved:** This metric represents the amount of memory saved in the data storage by representing the trajectory in the compressed format and it is calculated as:

[9] M is the number of states, G represents the number of unique observations (N) which does not go beyond the grid size, and D is the maximum state duration

$$MemorySaved = 1 - \left(\frac{compressed\ size}{uncompressed\ size}\right) \quad (6.8)$$

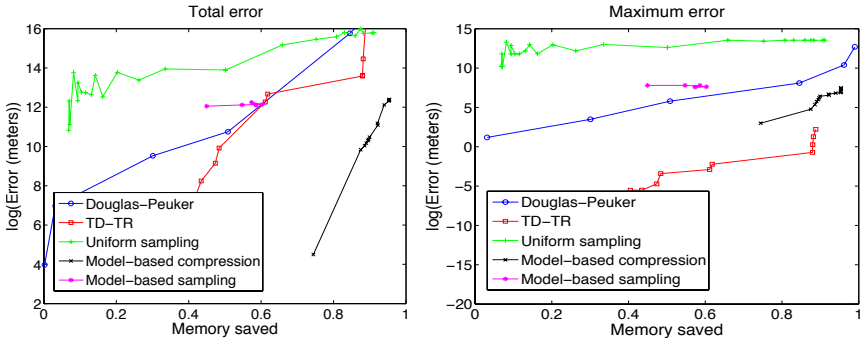
6.7.1 Case study using Dataset 1

Figure 6-1 compares the previously mentioned algorithms when applied on Dataset 1. These algorithms are compared in terms of total and maximum error versus the amount of memory saved. It is desirable that the algorithms can save as much memory as possible by maintaining lower error.

As seen in Figure 6-1.a, for this dataset the model-based compression technique performs better than the rest. Compared to the other algorithms, by saving higher amount of memory, this technique is resulting in a much lower amount of total error. The total error of TD-TR, Douglas-Peuker, and model-based sampling are very close to each other. Using model-based adaptive sampling the amount of memory saved is very limited to specific ranges. The reason is that, the other three compression algorithms (TD-TR, Douglas-Peuker, and model based compression) sample all the data and compress them afterwards. Adaptive sampling however, estimates the number of samples required and then acquires them. This way, compression is done before sampling. In order to have samples form start, middle, and end of each state, it is inevitable to have bounds for the amount of memory saved. However, it is seen that with a sampling scheme, which is based on the movement model, more than half of the samples are reduced. This is acquired with an error in the same range as TD-TR and Douglas-Peuker. Among the 5 techniques, which are compared here, Uniform Sampling is the worst in terms of total error. By using this technique even when the memory saved is very low, the error is still considerably high. Moreover, in lower compression ranges, instability in the error is observed. This is due to the fact that, the points are not chosen based on any significance criteria. Having minimal logic behind sampling, in some cases, highly significant points are removed leaving a considerable change in the total error.

Figure 6-1.b compares algorithms mentioned in Section 6.5.2 in terms of their maximum error versus the memory saved. The maximum error presented in this figure as mentioned in Section 6.6, is based on SED. It can be seen that, TD-TR followed by model-based compression algorithm performs considerably better than the other techniques in terms of maximum error. Better performance of TD-TR is expected as this compression algorithm operates based on SED. Model based compression is following TD-TR by having lower amount of maximum error. Higher error of Douglas-Peuker than model-based compression and TD-TR is justified by the fact that the error function in this algorithm is purely spatial (as opposed to SED which is spatio-temporal). In this case, performance of Douglas-Peuker is close to model-based sampling. As expected, uniform sampling has the highest

maximum error, which is consistently very high even when the amount of memory saved by this technique is very low.

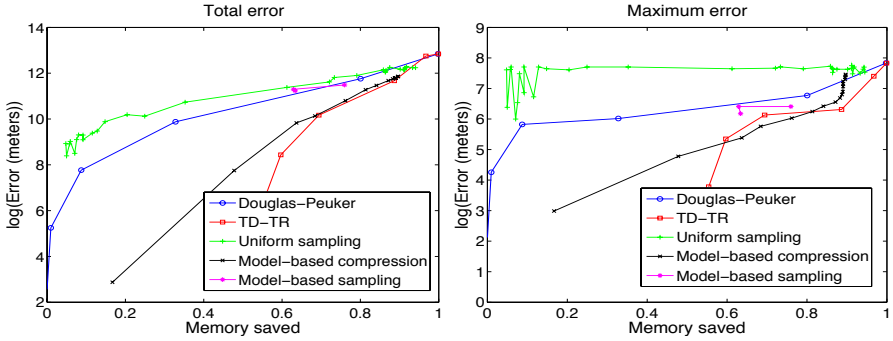


(a) (b)

Figure 6-1 Comparison of different methods in terms of memory saved versus, (a) total error, (b) maximum error (Dataset 1)

6.7.2 Case study using Dataset 2

The results of comparing compression algorithms on Dataset 2 (capricorns) is presented in Figure 6-2. In Figure 6-2.a the amount of total error versus memory saved is shown. As seen, the total error of TD-TR and model-based compression techniques are only marginally different. Douglas-Peuker and model-based sampling have the same total error and as expected uniform sampling has the highest amount of error compared with the rest of algorithms.



(a) (b)

Figure 6-2 Comparison of different methods in terms of memory saved versus, (a) total error, (b) maximum error (Dataset 2)

Figure 6-2.b represents the amount of memory saved versus the maximum error. As suggested by this figure, the three compression algorithms, Douglas-Peuker, model-based compression, and TD-TR are performing in the same range of maximum error. The total error of model-based sampling is slightly higher than the previously mentioned algorithms. Compared to the previous dataset, the maximum error is in general considerably higher for all methods. This similarity in performance and the higher maximum error can be justified by three reasons. These reasons are (i) animal's random movement, (ii) its limited movement range, and (iii) the fact that this dataset is already sparsely sampled. Therefore, these algorithms cannot represent their highest performance capacity in trajectory compression. As expected, the uniform sampling technique is performing worse than the other techniques and the difference between the maximum error resulted by this algorithm and that of the others, is considerably high.

6.6.1 Comparisons in terms of energy consumption

Contribution of different parts of a mobile sensor node in power consumption is different. Before we continue with comparing algorithms in terms of their energy consumption, we take a look at major energy consuming components of a basic mobile sensor node. Table 6-2 represents the amount of energy spent in radio and GPS sensor of a prototype mobile wireless sensor node.

	Component	Current (mA)	Contribution in sec/min	Energy in mWh
Measurement inactive	Radio TX	30	0.12	0.3
	Radio RX	8	0.1	0.07
Measurement active	Radio TX	30	0.32	0.8
	Radio RX	8	0.2	0.13
	GPS (POT high)	49	12	49
	GPS (POT low)	22	48	88

Table 6-2 Energy consumption in the radio and GPS components of mobile sensor node [235]

As seen in this table, when the sensor node is actively sensing GPS samples, the power consumption of the sensor is extremely higher than when it is inactive.

Figure 6-3 compares the algorithms mentioned in terms of the energy consumption of a typical mobile sensor node for sensing and transmitting previous datasets. For

each dataset, we have estimated the total energy consumption of the sensor node when running the algorithm against the total amount of space saved by compressing the trajectory. As expected, methods that encompass some form of reduction in sampling, such as uniform sampling and model based sampling will perform considerably better than the rest. Using these algorithms energy consumption is considerably low when the trajectory is highly compressed. Compared to these techniques, reduction in energy consumption of other compression algorithms (Douglas-Peuker, TD-TR, and Model based compression) is negligible. This is caused by the relatively high share of GPS component in total power consumption. Although compressed trajectory can reduce the amount of energy spend by the radio, this improvement is not visible. The major advantage of model-based sampling to uniform sampling is its operation with much lower amount of total and maximum error.

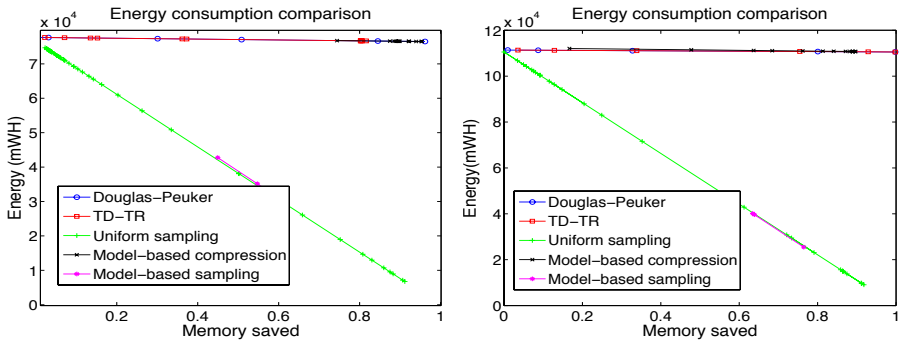


Figure 6-3 Comparison of different methods in terms of energy saved in the hardware versus space saved, (a) Dataset 1, (b) Dataset 2

6.7 Summary

In this chapter, we looked at a specific problem in mobility data sensing which is trajectory compression. Trajectories sensed with location acquisition devices can be extremely redundant specifically when inspected in temporal domain. In order to avoid sensing and transferring trajectories with such redundancies, we proposed using the state space models, which were studied before in Section 4.4. Firstly, we used the model to reduce the amount of data sent form sensor nodes to the base station based on their prediction of the next data point. Later, we extended the system using the model in an adaptive sampling technique. Results of comparison with state of the art techniques in trajectory compression show that the model based trajectory compression can effectively reduce the size of trajectories,

specifically when trajectories have repetitive patterns. Trading off for accuracy, the model based adaptive sampling can greatly reduce the power consumption of the sensor node and improve its lifetime. In this chapter, we studied model-based compression only using GPS data. Combining other sensors input such as accelerometer can also improve the performance of these compression algorithms.

Chapter 7

Conclusions and future directions

In this thesis we presented solutions for understanding the individual and social behavior of mobile entities from the mobility data. We addressed the requirements of such data analysis systems considering both application requirements and the constraints imposed by technology. In Chapter 2, we firstly reviewed different technologies, which can be used for collecting spatio-temporal data and compared them in terms of different performance parameters. In Chapters 3-5 we presented our contributions in understanding mobile entity's behavior from application point of view. In Chapter 6 we elaborated our contribution in mobile trajectory compression and transmission for resource-constrained devices to meet the requirements enforced by data collection technologies.

In this chapter, we review our contributions and results. Next, we present our lessons learnt and future direction.

7.1 Contributions

- **(Contribution 1) A review of Technological solutions for collecting spatio-temporal data from mobile objects**
  In order to choose the proper technology for collecting relevant data for our spatio-temporal data analysis, we reviewed different technological solutions. We classified these technologies based on their relevant application in movement modeling to two groups of Lagrangian and Eulerian technologies. Having reviewed the technologies, and their previous usage in mobile entity sensing application, we chose the Lagrangian based technologies and more specifically GPS data to proceed with our mobility data analysis since the technologies in the Eulerian category are yet far from usage in remote fields.
- **(Contribution 2) Extracting periodic patterns**
  In Chapter 3, we studied the problem of extracting periodic patterns from trajectories. We proposed a periodicity detection technique for streaming mobility data, which can extract periodic patterns with bounded memory and processing requirements. We evaluated the performance of this technique in presence of common uncertainties and showed that compared with the popularly used Autocorrelation function, the method performs with higher accuracy in presence of noise and uncertainties. Experiments

on different datasets show how periodic patterns can be accurately extracted and represent the mobile entities' periodic behavior.

- **(Contribution 3) Modeling trajectory dynamics**
  In Chapter 4, we proposed two trajectory-modeling techniques for understanding trajectory dynamics. We proposed both deterministic and probabilistic modeling solutions based on two different views on trajectories, which could capture short-term dependencies in the mobility data.
  In our deterministic modeling approach we segmented trajectories to find the frequently visited paths by mobile objects to the finest level of similarity. We compared performance of this technique with the other trajectory clustering approaches and showed that, using the collective knowledge of trajectories this technique can better perform in existence of uncertainties.
  In our probabilistic modeling approach we used hidden semi-Markov model, to model trajectories. Our proposed technique hierarchically models activities of the mobile entity in terms of states with their specific duration distribution. We compared this technique with other spatio-temporal models and showed that it can perform better in prediction of future mobility patterns. As the results on different mobility datasets show, complex patterns such as weekly patterns, long travel sequence, and change in the behaviors were all found using the second model.
  Finally, we compared both of these techniques and concluded that the first modeling approach can find frequent patterns efficiently and without extra time spent on interpreting the results. The second approach on the other hand, can discover unknown patterns at the expense of more effort by the data analyst in interpreting the results.
- **(Contribution 4) Extracting social context from mobility data**
  In Chapter 5, we addressed the interaction between entities in terms of the information extracted from mobility data and co-location patterns. We proposed two indicators based on information theory, which could focus on correlation in visits to frequent, and infrequent places. Using the MDC dataset, we studied the relationship between these indicators and the indicators extracted from phone call and sms features. The results show that these two groups of indicators are related and can be used as a basis for extracting social tie information. Also, compared with the commonly used co-location indicator, the proposed indicators perform better in representing the strength of social ties extracted from phone/sms indicators. Our case studies showed how these indicators represent information about the social ties between entities.

- **(Contribution 5) Model based trajectory sensing and compression**
  In order to increase the lifetime of the mobile device, and dealing with the ever-increasing loads of mobility data, in Chapter 5, we proposed two trajectory compression solutions. We used the trajectory modeling techniques proposed in Chapter 4.3 to (i) compress trajectories and (ii) to adaptively sample the data when it is mostly needed. The results of our experiments on two mobility datasets in this thesis shows that, (i) the proposed trajectory compression technique can perform superiorly in terms of total error, compression ratio, and energy preservation and (ii) although the adaptive trajectory sampling technique cannot meet optimum error requirements, it still performs considerably better than the unwise adaptive sampling while saving a considerable amount of energy.

7.2 Conclusions and Lessons learnt

The important lessons learnt during this thesis can be summarized as:

- **The rate of data generation is so high that there is no option other than streaming data analysis**
  Being able to analyze streams of data as they arrive is a prerequisite of various real time applications. Nevertheless, with the massive data explosion and the increasing rate of data generation of any kind, it is evident that saving incoming data for future analysis is not an option anymore even for non-real time applications.
- **The paradox of incompleteness in excess**
  The question always at back of our minds during all chapters of this thesis was: "*How to deal with missing data*?". At the same time, there is so much redundancy in the mobility data in terms of short-term, long-term, and social patterns that the missing data do not necessarily imply loss of content anymore. The previous statement holds true so long as, there is an efficient inference technique, which uses these redundancies to fill the lack of data.
- **Looking at problems from different perspectives provides advantage**
  In Chapter 4, we addressed the problem of modeling trajectories from two perspectives. One based on intuition and the other based on mathematics. Eventually, we cannot say which one is the best but both can be used for the relevant application.
- **Worry or enthusiasm?**
  Last but not least, talking about data analysis, the first reaction is always to worry. Mobility data, as shown in this thesis are rich in different contextual information. The applications are plentiful but so are the threats. Rather than impeding the flow of data, there should be actions taken to avoid the possible threats.

7.3 Future research directions

In this thesis a number of research questions regarding mobility data analysis were studied. There are still issues to be addressed in future research. Some of these issues are:

- **Support for streaming data:** We addressed the problem of mining streaming data in Chapter 3. However, there is still more effort required in this domain. For instance, in Chapter 4 we proposed trajectory-modeling approaches, which are not yet supporting streaming data. Providing support for streaming version of algorithms proposed in Chapter 4 is beneficial for future applications.
- **Group behavior and community detection:** In Chapter 5, we studied social context from mobility data. Another interesting topic not yet well studied is community detection and mining group behavior from such data.
- **Incorporation of other types of data:** In this thesis, we mainly focused on mobility data acquired from GPS. Many other types of data, which are less studied, can also be topic of research using similar techniques. For instance, the data collected by Wi-Fi access points, or PIR sensors can be used to provide finer-grained information about habits and interactions.
- **Finding "abnormal" behavior using streams of mobility data**: In different chapters of this thesis (Chapter 3,4) we showed how we could use different techniques to identify frequent or normal behavior. Another research direction can be using these techniques to also see how we can identify abnormal behavior.
- **More application specific research:** There are many applications which can benefit from knowledge hidden in data. The data analysis results can be directed into usage to acquire knowledge of matters of high importance such as: "How livable cities are?" or "How to deal with disasters?".
- **Support for maximum security:** As mentioned in the previous section the threats of mishandling data analysis techniques are numerous. Mobility data can explain a lot about the mobile entity. When, social context is available, the social network of the mobile entity can reveal the identity of the mobile entity. Therefore, simple anonymization of mobility data does not provide maximum privacy. Looking for privacy techniques, which ensures only positive results from data analysis, is an important research direction.

List of publications

Published:

- M. Baratchi, N. Meratnia, P.J.M. Havinga, A.K. Skidmore, and A.G. Toxopeus, "Sensing solutions for collecting spatio-temporal data for wildlife monitoring applications: a review," Sensors, vol. 13, pp. 6054-6088, 2013.
- M. Baratchi, N. Meratnia, and P.J.M. Havinga, "Finding frequently visited paths: dealing with the uncertainty of spatio-temporal mobility data," in Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP'13), 2013 IEEE Eighth International Conference on, 2013, pp. 479-484.
- M. Baratchi, N. Meratnia, and P.J.M. Havinga, "Recognition of periodic behavioral patterns from streaming mobility data," in proceedings of Mobile and Ubiquitous Systems: Computing, Networking, and Services (MOBIQUITOUS'13), ed: Springer International Publishing, 2014, pp. 102-115.
- M. Baratchi, N. Meratnia, and P.J.M. Havinga, "On the use of mobility data for discovery and description of social ties," in Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM'13), 2013, pp. 1229-1236.
- M. Baratchi, N. Meratnia, P.J.M. Havinga, A.K. Skidmore, and A.G. Toxopeus, "A hierarchical hidden semi-Markov model for modeling mobility data," in proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (Ubicomp'14), 2014, pp. 401-412.

Manuscript in process:

- M. Baratchi, N. Meratnia, P.J.M. Havinga, A.K. Skidmore, A.G. Toxopeus, "Discovery and description of social ties from mobility data," Manuscript in process for submission.
- M. Baratchi, N. Meratnia, P.J.M. Havinga, A.K. Skidmore, A.G. Toxopeus, "Model-based trajectory compression and adaptive sampling," Manuscript in process for submission.

Bibliography

[1] IBM. (2014). Available: http://www-01.ibm.com/software/data/bigdata/what-is-big-data.html

[2] H. L. Beyer and D. T. Haydon, "The interpretation of habitat preference metrics under useavailability designs," *Phil. Trans. R. Soc. B,* p. 7, 2010.

[3] N. Owen-Smith, J. M. Fryxell, and E. H. Merrill, "Foraging theory upscaled: the behavioural ecology of herbivore movement," *Phil. Trans. R. Soc. B,* p. 12, 2010.

[4] M. Valeix, A. J. Loveridge, S. Chamaillé-Jammes, Z. Davidson, F. Murindagomo, H. Fritz*, et al.*, "Behavioral adjustments of African herbivores to predation risk by lions: Spatiotemporal variations influence habitat use," *Ecology,* vol. 90, pp. 23-30, 2009/01/01 2009.

[5] M. J. Chamberlain and B. D. Leopold, "Spatio-temporal Relationships Among Adult Raccoons (Procyon lotor) in Central Mississippi," *The American Midland Naturalist,* vol. 148, pp. 297-308, 2002/10/01 2002.

[6] G. Adomavicius and A. Tuzhilin, "Context-Aware Recommender Systems," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds., ed: Springer US, 2011, pp. 217-253.

[7] S. Aflaki, N. Meratnia, M. Baratchi, and P. J. M. Havinga, "Evaluation of Incentives for Body Area Network-based HealthCare Systems," presented at the IEEE ISSNIP, Melbourne, Australia, 2013.

[8] R. K. Rana, C. T. Chou, S. S. Kanhere, N. Bulusu, and W. Hu, "Ear-phone: an end-to-end participatory urban noise mapping system," presented at the Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, Stockholm, Sweden, 2010.

[9] C. Ratti, S. Williams, D. Frenchman, and R. Pulselli, "Mobile landscapes: using location data from cell phones for urban analysis," *Environment and Planning B Planning and Design,* vol. 33, p. 727, 2006.

[10] L. Bengtsson, X. Lu, A. Thorson, R. Garfield, and J. Von Schreeb, "Improved response to disasters and outbreaks by tracking population movements with mobile phone network data: a post-earthquake geospatial study in Haiti," *PLoS medicine,* vol. 8, p. e1001083, 2011.

[11] (2014). *Mobile Wireless Sensor Networks.* Available: http://en.wikipedia.org/wiki/Mobile_wireless_sensor_network

[12] D. Pfoser and C. Jensen, "Capturing the Uncertainty of Moving-Object Representations Advances in Spatial Databases." vol. 1651, R. Güting, D. Papadias, and F. Lochovsky, Eds., ed: Springer Berlin / Heidelberg, 1999, pp. 111-131.

[13] C. Katsaounis, "Habitat use of the endangered and endemic cretan capricorn and impact of domestic goats," Master of Science, University of Twente Faculty of Geo-Information and Earth Observation, Enschede, 2012.

[14] M. C. Nicholson and T. P. Husband, "DIURNAL BEHAVIOR OF THE AGRIMI, CAPRA-AEGAGRUS," *Journal of Mammalogy,* vol. 73, pp. 135-142, Feb 1992.

[15] Yu Zheng, Xing Xie, and W.-Y. Ma, "GeoLife: A Collaborative Social Networking Service among User, location and trajectory," *IEEE Data(base) Engineering Bulletin,* 2010.

[16] Y. Yingxiang, "Understanding human mobility patterns from digital traces," Master of science, Civil and environmental engineering, Massachusettes institute of technology, 2011.

[17] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, "Mining interesting locations and travel sequences from GPS trajectories," presented at the Proceedings of the 18th international conference on World wide web, Madrid, Spain, 2009.

[18] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W.-Y. Ma, "Understanding mobility based on GPS data," in *Proceedings of the 10th international conference on Ubiquitous computing*, 2008, pp. 312-321.

[19] J. K. Laurila, D. Gatica-Perez, I. Aad, B. J, O. Bornet, T.-M.-T. Do*, et al.*, "The Mobile Data Challenge: Big Data for Mobile Computing Research," presented at the Pervasive Computing, Newcastle, 2012.

[20] N. Kiukkonen;, O. D. J. Blom, and a. J. L. D. Gatica-Perez, "Towards rich mobilephone datasets: Lausanne data collection campaign," in *CPS, Berlin* 2010.

[21] M. Baratchi, N. Meratnia, P. J. M. Havinga, A. K. Skidmore, and A. G. Toxopeus, "Sensing solutions for collecting spatio-temporal data for wildlife monitoring applications: a review," *Sensors,* vol. 13, pp. 6054-6088, 2013.

[22] P. E. Smouse, S. Focardi, P. R. Moorcroft, J. G. Kie, J. D. Forester, and J. M. Morales, "Stochastic modelling of animal movement," *Philosophical Transactions of the Royal Society B: Biological Sciences,* vol. 365, pp. 2201-2211, July 27, 2010 2010.

[23] C. J. Baker, M. Vespe, and G. J. Jones, "Target classification by echo locating animals," presented at the Waveform Diversity and Design Conference, 2007. International, Pisa, Italy, 2007.

[24] P. K. Dutta, A. K. Arora, and S. B. Bibyk, "Towards radar-enabled sensor networks," presented at the Proceedings of the 5th international conference on Information processing in sensor networks, Nashville, Tennessee, USA, 2006.

[25] Samraksh. (2008, 2011.09.10). *Users Manual for the BumbleBee (Model 0).* Available: http://www.samraksh.com

[26] D. Tahmoush and J. Silvious, "Angle, elevation, PRF, and illumination in radar microDoppler for security applications," in *Antennas and Propagation Society International Symposium, 2009. APSURSI '09. IEEE*, 2009, pp. 1-4.

[27] S. Thayaparan, S. Abrol, and E. Riseborough, "Micro-Doppler radar signatures for intelligent target recognition," Ottawa2004.

[28] J. L. Geisheimer, E. F. Greneker III, and W. S. Marshall, "High-resolution Doppler model of the human gait," in *SPIE 4744*, 2002.

[29] D. Tahmoush and J. Silvious, "Radar micro-doppler for long range front-view gait recognition," in *Biometrics: Theory, Applications, and Systems, 2009. BTAS '09. IEEE 3rd International Conference on*, Adelphie, MD, USA, 2009, pp. 1-6.

[30] Z. Zhaonian and A. G. Andreou, "Human identification experiments using acoustic micro-Doppler signatures," in *Micro-Nanoelectronics, Technology and Applications, 2008. EAMTA 2008. Argentine School of*, Washington, DC, USA, 2008, pp. 81-86.

[31] Z. Qin, L. Jianhan, A. Host-Madsen, O. Boric-Lubecke, and V. Lubecke, "Detection of multiple heartbeats using doppler radar," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Tolouse, France, 2006.

[32] Z. Zhang, P. Pouliquen, A. Waxman, and A. G. Andreou, "Acoustic Micro-Doppler gait signatures of humans and animals," in *Information Sciences and Systems, 2007. CISS '07. 41st Annual Conference on*, Baltimore, MD, USA, 2007, pp. 627-630.

[33] S. G. Zaugg S., van Loon E., Schmaljohann H., Liechti F., "Automatic identification of bird targets with radar via patterns produced by wing flapping," *Journal of Royal Society Interface,* vol. 5, 2008.

[34] A. M. Dokter, F. Liechti, H. Stark, L. Delobbe, P. Tabary, and I. Holleman, "Bird migration flight altitudes studied by a network of operational weather radars," *Jornal of Royal Society Interface,* vol. 8, pp. 30-43, 2010.

[35] J. M. Ruth, W. C. Barrow, R. S. Sojda, D. K. Dawson, R. H. Diehl, A. Manville*, et al.*, "Using radar to advance migratory bird management: an interagency collaboration," USGS Fort Collins Science Center, Reston, VA, USA, Geological Survey Fact Sheet 2005-30482005.

[36] K. J. Benoit-Bird, W. W. L. Au, C. D. Kelley, and C. Taylor, "Acoustic backscattering by deepwater fish measured in situ from a manned submersible," *Deep Sea Research Part I: Oceanographic Research Papers,* vol. 50, pp. 221-229, 2003.

[37] K. Youngwook and L. Hao, "Human activity classification based on Micro-Doppler signatures using a Support Vector Machine," *Geoscience and Remote Sensing, IEEE Transactions on,* vol. 47, pp. 1328-1337, 2009.

[38] O. R. Fogle, "Human micro-ranfe/micro-doppler signature signature extraction, association, and statistical characterization for high-resolution radar," PhD, Wright State University, 2011.

[39] L. N. Anishchenko, A. S. Bugaev, S. I. Ivashov, and V. I. A., "Application of Bioradiolocation for Estimation of the Laboratory Animals' Movement Activity," *PIERS Online,* vol. 5, 2009.

[40] M. Rahimi, R. Baer, O. I. Iroezi, J. C. Garcia, J. Warrior, D. Estrin*, et al.*, "Cyclops: in situ image sensing and interpretation in wireless sensor networks," presented at the Proceedings of the 3rd international conference on Embedded networked sensor systems, San Diego, California, USA, 2005.

[41] (3/5/2013). *CMUcam: Open Source Programmable Embedded Color Vision Sensors*. Available: http://www.cmucam.org/

[42] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu, "SensEye: a multi-tier camera sensor network," presented at the Proceedings of the 13th annual ACM international conference on Multimedia, Hilton, Singapore, 2005.

[43] M. Lahiri, C. Tantipathananandh, R. Warungu, D. I. Rubenstein, and T. Y. Berger-Wolf, "Biometric animal databases from field photographs: identification of individual zebra in the wild," presented at the Proceedings of the 1st ACM International Conference on Multimedia Retrieval, Trento, Italy, 2011.

[44] L.-F. Liu, W. Jia, and Y.-H. Zhu, "Survey of Gait Recognition," in *Emerging Intelligent Computing Technology and Applications. With Aspects of Artificial Intelligence.* vol. 5755, D.-S. Huang, K.-H. Jo, H.-H. Lee, H.-J. Kang, and V. Bevilacqua, Eds., ed: Springer Berlin Heidelberg, 2009, pp. 652-659.

[45] J. B. Hayfron-Acquah, M. S. Nixon, and J. N. Carter, "Recognising human and animal movement by symmetry," in *International Conference on Image Processing*, Brussels, Belgium, 2001.

[46] S. Mimura, K. Itoh, T. Kobayashi, T. Takigawa, A. Tajima, A. Sawamura*, et al.*, "The cow gait recognition using CHLAC," presented at the Bio-inspired Learning and Intelligent Systems for Security, BLISS '08. ECSIS Symposium on, Edinburgh, UK, 2008.

[47] D. Tweed and A. Calway, "Tracking multiple animals in wildlife footage," in *Pattern Recognition, Proceedings. 16th International Conference on*, Quebec, QC, Canada, 2002, pp. 24-27 vol.2.

[48] S. L. Hannuna, N. W. Campbell, and D. P. Gibson, "Segmenting quadruped gait patterns from wildlife video," in *IEE Conference Publications*, 2005, pp. 235-242.

[49] T. Burghardt and J. Calic, "Analysing animal behaviour in wildlife videos using face detection and tracking," *Vision, Image and Signal Processing, IEE Proceedings -,* vol. 153, pp. 305-312, 2006.

[50] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.,* vol. 38, p. 13, 2006.

[51] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Comput. Surv.,* vol. 43, pp. 1-43, 2011.

[52] S. Belongie, K. Branson, P. Dollár, and V. Rabaud. (2005, Monitoring animal behavior in the smart vivarium. 70-72.

[53] W. Gonçalves, J. de Andrade Silva, B. Machado, H. Pistori, and A. de Souza, "Hidden Markov models applied to snakes behavior identification advances in image and video technology." vol. 4872, D. Mery and L. Rueda, Eds., ed: Springer Berlin / Heidelberg, 2007, pp. 777-787.

[54] T. S. Group. (2013, 23 April 2013). Available: http://www.thesnellgroup.com

[55] M. Stewart, "Non-Invasive Measurement of Stress and Pain in Cattle Using Infrared Thermography," Massey University, Palmerston North, New Zealand, 2009.

[56] R. Szewczyk, J. Polastre, A. Mainwaring, and D. Culler, "Lessons from a sensor network expedition wireless sensor networks." vol. 2920, H. Karl, A. Wolisz, and A. Willig, Eds., ed: Springer Berlin / Heidelberg, 2004, pp. 307-322.

[57] A. W. Claridge, G. Mifsud, J. Dawson, and M. J. Saxon, "Use of infrared digital cameras to investigate the behaviour of cryptic species," *Wildlife Research,* vol. 31, pp. 645-650, 2004.

[58] M. Tahir, P. Hung, R. Farrell, and S. McLoone, "Lightweight signal processing algorithms for human activity monitoring using dual PIR-sensor nodes," in *CIICT 2009 : proceedings of the China-Ireland information and communications technologies conference*, Ireland, 2009, pp. 150-156.

[59] W. Weihong, Z. Jian, and S. Chunhua, "Improved human detection and classification in thermal images," in *Image Processing (ICIP), 17th IEEE International Conference on,* Seattle, WA, USA, 2010, pp. 2313-2316.

[60] P. Buddharaju, I. T. Pavlidis, P. Tsiamyrtzis, and M. Bazakos, "Physiology-based face recognition in the thermal infrared spectrum," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 29, pp. 613-626, 2007.

[61] M. Andersson, J. Rydell, and J. Ahlberg, "Estimation of crowd behavior using sensor networks and sensor fusion," in *Information Fusion, FUSION '09. 12th International Conference on*, 2009, pp. 396-403.

[62] M. Stewart, J. R. Webster, A. L. Schaefer, N. J. Cook, and S. L. Scott, "Infrared thermography as a non-invasive tool to study animal welfare," *Animal welfare,* pp. 319-325, 2005.

[63] C. Lavers and et al., "Application of remote thermal imaging and night vision technology to improve endangered wildlife resource management with minimal animal distress and hazard to humans," *Journal of Physics: Conference Series,* vol. 15, pp. 207-212, 2005.

[64] J. Gloster, K. Ebert, S. Gubbins, J. Bashiruddin, and D. Paton, "Normal variation in thermal radiated temperature in cattle: implications for foot-and-mouth disease detection," *BMC Veterinary Research,* vol. 7, pp. 73-84, 2011.

[65] F. E. Regnier and J. H. Law, "Insect pheromones," *Journal of Lipid Research,* vol. 9, pp. 541-551, September 1, 1968 1968.

[66] J. Qu, Y. Chai, and S. Yang, "A Real-Time De-Noising Algorithm for E-Noses in a Wireless Sensor Network," *Sensors,* vol. 9, pp. 895-908, 2009.

[67] R. A. Ramadan, "Odor Recognition and Localization Using Sensor Networks," in *Wireless Sensor Networks: Application-Centric Design*, Y. K. Tan, Ed., ed, 2010.

[68] Y. Kuwana, S. Nagasawa, I. Shimoyama, and R. Kanzaki, "Synthesis of the pheromone-oriented behaviour of silkworm moths by a mobile robot with moth antennae as pheromone sensors," *Biosensors and Bioelectronics,* vol. 14, pp. 195-202, 1999.

[69] Y. Lan, X. Zheng, J. K. Westbrook, J. Lopez, R. Lacey, and W. C. Hoffmann, "Identification of Stink Bugs Using an Electronic Nose," *Journal of Bionic Engineering,* vol. 5, Supplement, pp. 172-180, 2008.

[70] W. G. Henderson, A. Khalilian, Y. J. Han, J. K. Greene, and D. C. Degenhardt, "Detecting stink bugs/damage in cotton utilizing a portable electronic nose," *Comput. Electron. Agric.,* vol. 70, pp. 157-162, 2010.

[71] (10 April 2013). *Cyranose.* Available: http://www.ideo.com/work/cyranose-320

[72] M. Ciganek and J. Neca, "Chemical characterization of volatile organic compounds on animal farms," *Veterinarni Medicina,* vol. 53, pp. 641-651, 2008.

[73] P. Leilei and S. X. Yang, "An electronic nose network system for online monitoring of livestock farm odors," *Mechatronics, IEEE/ASME Transactions on,* vol. 14, pp. 371-376, 2009.

[74] C. Wongchoosuk, M. Lutz, and T. Kerdcharoen, "Detection and classification of human body odor using an electronic nose," *Sensors,* vol. 9, pp. 7234-7249, 2009.

[75] P. A. Brennan and K. M. Kendrick, "Mammalian social odours: attraction and individual recognition," vol. The Royal society, pp. 2061-2078, 2006.

[76] C. D. Buesching, J. S. Waterhouse, and D. W. Macdonald, "Gas-Chromatographic Analyses of the Subcaudal Gland Secretion of the European Badger (Meles meles)

Part II: Time-Related Variation in the Individual-Specific Composition," *Journal of Chemical Ecology,* vol. 28, pp. 57-69, 2002.

[77] S. Fuchs, P. Strobel, M. Siadat, and M. Lumbreras, "Evaluation of unpleasant odor with a portable electronic nose," *Materials Science and Engineering: C,* vol. 28, pp. 949-953, 2008.

[78] D. T. Blumstein, D. J. Mennill, P. Clemins, L. Girod, K. Yao, G. Patricelli*, et al.*, "Acoustic monitoring in terrestrial environments using microphone arrays: applications, technological considerations and prospectus," *Journal of Applied Ecology,* vol. 48, pp. 758-767, 2011.

[79] TinyOsGroup. (2011, 10 April 2013). *TinyOS developers guide*. Available: http://www.cs.wmich.edu/wsn/doc/micasbl.pdf

[80] L. Girod, M. Lukac, V. Trifa, and D. Estrin, "The design and implementation of a self-calibrating distributed acoustic sensing platform," presented at the Proceedings of the 4th international conference on Embedded networked sensor systems, Boulder, Colorado, USA, 2006.

[81] J. Hershberger and J. Snoeyink, "Cartographic line simplification and polygon CSG formulæ in O(n log∗ n) time," *Computational Geometry,* vol. 11, pp. 175-185, 12// 1998.

[82] E. B. Crouch and P. W. C. Paton, "Assessing the Use of Call Surveys to Monitor Breeding Anurans in Rhode Island," *Journal of Herpetology,* vol. 36, pp. 185-193, 2002.

[83] M. C. MacSwiney G, F. M. Clarke, and P. A. Racey, "What you see is not what you get: the role of ultrasonic detectors in increasing inventory completeness in Neotropical bat assemblages," *Journal of Applied Ecology,* vol. 45, pp. 1364-1371, 2008.

[84] S. Shukla, N. Bulusu, and S. Jha, "Cane-toad Monitoring in Kakadu National Park Using Wireless Sensor Networks," in *APAN*, Carnes,Tarcoola, Australia, 2004.

[85] S. Fagerlund, "Bird Species Recognition Using Support Vector Machines," *EURASIP Journal on Advances in Signal Processing,* vol. 07, pp. 64-71, 2007.

[86] J. Cai, D. Ee, P. Binh, P. Roe, and J. Zhang, "Sensor network for the monitoring of ecosystem: bird species recognition," in *Intelligent Sensors, Sensor Networks and Information, 2007. ISSNIP 2007. 3rd International Conference on*, Brisbane, QLD, Australia, 2007, pp. 293-298.

[87] B. Wessling, "Individual recognition of cranes, monitoring and vocal communication analysis by sonography," in *4th European crane workshop*, Fenetrange, France, 2000.

[88] S. Hartwig, "Individual acoustic identification as a non-invasive conservation tool: An approach to the conservation of the African wild dog Lycaon pictus (Temminck, 1820)." *Bioacoustics The International Journal Of Animal Sound And Its Recording,* vol. 15, pp. 35-50, 2005.

[89] T. Grava, N. Mathevon, E. Place, and P. Balluet, "Individual acoustic monitoring of the European Eagle Owl Bubo bubo," *Ibis,* vol. 150, pp. 279-287, 2008.

[90] A. N. G. Kirschel, M. L. Cody, Z. T. Harlow, V. J. Promponas, E. E. Vallejo, and C. E. Taylor, "Territorial dynamics of Mexican Ant-thrushes Formicarius moniliger revealed by individual recognition of their songs," *Ibis,* vol. 153, pp. 255-268, 2011.

[91] S. G. Iyengar, P. K. Varshney, and T. Damarla, "On the detection of footsteps based on acoustic and seismic sensing," in *Signals, Systems and Computers. ACSSC 2007.*

Conference Record of the Forty-First Asilomar Conference on, Pacific Grove, CA, USA, 2007, pp. 2248-2252.

[92] A. Itai and H. Yasukawa, "Footstep Classification Using Wavelet Decomposition," presented at the Communications and Information Technologies, 2007. ISCIT '07. International Symposium on, Sydney, NSW, Australia, 2007.

[93] G. Sleife, M. D. Ladd, T. S. McDonald, and G. E. Sleefe, *Acoustic and Seismic Modalities for Unattended Ground Sensors*. Orlando, FL. USA, 1999.

[94] H. O. Park, A. A. Dibazar, and T. W. Berger, "Cadence Analysis of Temporal Gait Patterns for Seismic Discrimination Between Human and Quadruped Footsteps," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009)*, Taipei, China, 2009, pp. 1749-1752.

[95] G. F. Miller, Pursey, H., "On the partition of energy between elastic waves in a semi-infinite solid," in *Roy. Soc. London, Series A*, 1955, pp. 55-69.

[96] G. E. Sleefe, M. D. Ladd, T. S. McDonald, and G. J. Elbring, "Acoustic and Seismic Modalities for Unattended Ground Sensors," in *SPIE*, Orlando, FL, USA, 1999.

[97] G. P. Mazarakis and J. N. Avaritsiotis, "A prototype sensor node for footstep detection," in *Wireless Sensor Networks. Proceeedings of the Second European Workshop on*, 2005, pp. 415-418.

[98] A. Pakhomov, A. Sicignano, Sandy., and A. Goldburt, " Seismic footstep signal characterization," *SPIE proceedings series* 2003.

[99] R. W. Mankin and J. Benshemesh, "Geophone detection of subterranean termite and ant activity," *Journal of Economic Entomology,* vol. 99, pp. 244-250, 2006/02/01 2006.

[100] E. S. Nadimi, H. T. Søgaard, and T. Bak, "ZigBee-based wireless sensor networks for classifying the behaviour of a herd of animals using classification trees," *Biosystems Engineering,* vol. 100, pp. 167-176, 2008.

[101] S. Oh, H. Kwon, H. Yoon, and V. K. Varadan, "Application of wireless sensor system on security network," in *SPIE smart*, 2010, pp. 76460-76460.

[102] V. S. Nithya, K. Sheshadri, A. Kumar, and K. V. S. Hari, "Model based target tracking in a wireless network of passive infrared sensor nodes," in *Signal Processing and Communications (SPCOM), 2010 International Conference on*, 2010, pp. 1-5.

[103] G. P. Succi, D. Clapp, R. Gampert, and G. Prado, "Footstep detection and tracking," in *SPIE*, 2001.

[104] Z. Liang, J. Wei, J. Zhao, H. Liu, B. Li, J. Shen*, et al.*, "The statistical meaning of kurtosis and Its new application to ientification of persons based on seismic signals," *Sensors,* vol. 8, pp. 5106-5119, 2008.

[105] A. Pakhomov and T. Goldburt, "New seismic unattended small size module for footstep and light and heavy vehicles detection and identification," presented at the SPIE, Orlando, FL, USA, 2007.

[106] K. M. Houston and D. P. McGaffigan, "Spectrum analysis techniques for personnel detection using seismic sensors," in *SPIE*, Orlando, FL, USA, 2003, pp. 162-173.

[107] A. Pakhomov, A. Sicignano, M. Sandy, and E. T. Goldburt, "Single and three-axis geophone: footstep detection with bearing estimation, localization, and tracking," presented at the SPIE, Orlando, FL, USA, 2003.

[108] S. Schumer, "Analysis of human footsteps utilizing multi-axial seismic fusion," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 697-700.

[109] J. D. Wood, C. E. O'Connell-Rodwell, and S. L. Klemperer, "Methodological insights: Using seismic sensors to detect elephants and other large mammals: a potential census technique," *Journal of Applied Ecology,* vol. 42, pp. 587-594, 2005.

[110] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik*, et al.*, "A line in the sand: a wireless sensor network for target detection, classification, and tracking," *Computer Networks,* vol. 46, pp. 605-634, 2004.

[111] P. Narins, O. J. Reichman, J. M. Jarvis, and E. Lewis, "Seismic signal transmission between burrows of the Cape mole-rat, Georychus capensis," *Journal of Comparative Physiology A,* vol. 170, pp. 13-21, 1992/01/01 1992.

[112] M. Chiesa, R. Genz, F. Heubler, K. Mingo, and C. Noessel, "RFID a week long survey on the technology and its potential Radio Frequency IDentification," 2005.

[113] P. H. Becker and H. Wendeln, "A new application for transponders in population ecology of the common tern," *The Condor,* vol. 99, 1997.

[114] A. W. Roark and M. E. Dorcas, "Regional body temperature variation in corn snakes measured using temperature-sensitive passive integrated transponders," *Journal of Herpetology,* vol. 34, p. 4, 2000.

[115] S. Hsi and H. Fait, "RFID enhances visitors' museum experience at the Exploratorium," *Commun. ACM,* vol. 48, pp. 60-65, 2005.

[116] U. Schulte, D. sters, and S. Steinfartz, "A PIT tag based analysis of annual movement patterns of adult fire salamanders (Salamandra salamandra) in a Middle European habitat," *Amphibia-Reptilia,* vol. 28, pp. 531-536, 2007.

[117] V. Dyo, S. A. Ellwood, D. W. Macdonald, A. Markham, C. Mascolo, B. Psztor*, et al.*, "Evolution and Sustainability of a Wildlife Monitoring Sensor Network," presented at the Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems, Zurich, Switzerland, 2010.

[118] P. Sikka, P. Corke, P. Valencia, C. Crossman, D. Swain, and G. Bishop-Hurley, "Wireless adhoc sensor and actuator networks on the farm," presented at the Proceedings of the 5th international conference on Information processing in sensor networks, Nashville, Tennessee, USA, 2006.

[119] M. Bouet and A. L. dos Santos, "RFID tags: Positioning Principles and Localization Techniques," in *Wireless Days (WD 08)*, Dubai, UAE, 2008, pp. 1-5.

[120] W. Ron. (2005) RFID: A Technical Overview and Its Application to the Enterprise. 27-33. Available: http://doi.ieeecomputersociety.org/10.1109/MITP.2005.69

[121] J. M. Roussel, A. Haro, and R. A. Cunjak, "Field test of a new method for tracking small fishes in shallow rivers using passive integrated transponder (PIT) technology," *Canadian Journal of Fisheries and Aquatic Sciences,* vol. 57, pp. 1326-1329, 2000/07/01 2000.

[122] E. J. H. Robinson, F. D. Smith, K. M. E. Sullivan, and N. R. Franks, "Do ants make direct comparisons?," *Proceedings of the Royal Society B: Biological Sciences,* vol. 276, pp. 2635-2641, July 22, 2009 2009.

[123] R. Weinstein, "RFID: a technical overview and its application to the enterprise," *IT Professional,* vol. 7, pp. 27-33, 2005.

[124] Biomark. (2012, 23 April 2013). Available: http://www.biomark.com/Products/Tags/Bulk_PIT_tags/

[125] Skyetek. (2012, 2012.3.5). *Skyetek*. Available: http://www.skyetek.com

[126] K. Thorup and R. A. Holland, "The bird GPS – long-range navigation in migrants," *Journal of Experimental Biology,* vol. 212, pp. 3597-3604, November 15, 2009 2009.

[127] B. J. M. Stutchbury, S. A. Tarof, T. Done, E. Gow, P. M. Kramer, J. Tautin*, et al.*, "Tracking long-distance songbird migration by using geolocators," *Science,* vol. 323, pp. 896-896, 2009.

[128] Televilt. (2013, 6 April). *Followit*. Available: http://www.followit.se/

[129] Norstar. (2013, 11 April). Available: http://www.northstarst.com/

[130] lotek. (2012, 2012.1.19). *Lotek wireless Inc*. Available: http://www.lotek.com/

[131] e-obs. (2013, 10 April). *e-obs*. Available: http://www.e-obs.de

[132] Microwave. (2013, 21 April). *Microwave Telemetry*. Available: http://www.microwavetelemetry.com/

[133] P. Sikka, P. Corke, and L. Overs, "Wireless Sensor Devices for Animal Tracking and Control," in *29th Annual IEEE International Conference on Local Computer Networks.* , Tampa, FL. USA, 2004, pp. 446-454.

[134] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet," *Acm Sigplan Notices,* vol. 37, pp. 96-107, Oct 2002.

[135] M. Rutishauser, V. Petkov, J. Boice, K. Obraczka, P. Mantey, T. M. Williams*, et al.*, "CARNIVORE: a disruption-tolerant system for studying wildlife," *EURASIP J. Wirel. Commun. Netw.,* vol. 11, pp. 1-14, 2011.

[136] J. H. Huang, Y. Y. Chen, Y. T. Huang, L. Po-Yen, C. Yi-Chao, L. Yi-Fu*, et al.*, "Rapid prototyping for wildlife and ecological monitoring," *Systems Journal, IEEE,* vol. 4, pp. 198-209, 2010.

[137] Z. Butler, P. Corke, R. Peterson, and D. Rus, "Networked Cows: Virtual Fences for Controlling Cows," presented at the IEEE International Conference on Robotics and Automation, 2004, New Orleans, Louisiana, USA, 2004.

[138] R. Handcock, D. Swain, G. Bishop-Hurley, K. Patison, T. Wark, P. Valencia*, et al.*, "Monitoring animal behaviour and environmental interactions using wireless sensor networks, GPS collars and satellite remote sensing," *Sensors,* vol. 9, pp. 3586-3603, 2009.

[139] Y. Xiaoping, E. R. Bachmann, H. Moore, and J. Calusdian, "Self-Contained Position Tracking of Human Movement Using Small Inertial/Magnetic Sensor Modules," in *Robotics and Automation, IEEE International Conference on*, Rome, Italy, 2007, pp. 2526-2533.

[140] F. Lei, P. J. Antsaklis, L. A. Montestruque, M. B. McMickell, M. Lemmon, S. Yashan*, et al.*, "Design of a wireless assisted pedestrian dead reckoning system - the NavMote experience," *Instrumentation and Measurement, IEEE Transactions on,* vol. 54, pp. 2342-2358, 2005.

[141] J. Thiele, O. Osechas, J. Bitsch, and K. Wehrle, "Smart Sensors for Small Rodent Observation," in *IEEE Sensors 2008* Lesse, Italy, 2008, pp. 709-711.

[142] J. Thiele, J. A. B. Link, and O. Osechas, "Dynamic wireless sensor networks for animal behavior research," in *New Developements in Biomedical Engineering*, ed Vienna, 2010.

[143] L. Bao and S. Intille, "Activity Recognition from User-Annotated Acceleration Data," in *Pervasive Computing*. vol. 30, A. Ferscha and F. Mattern, Eds., ed: Springer Berlin Heidelberg, 2004, pp. 1-17.

[144] E. Shepard, R. P. Wilson, F. Quintana, L. A. Gómez, N. Liebsch, D. A. Albareda, *et al.*, "Identification of animal movement patterns using tri-axial accelerometry," *Endangered Species Research,* vol. 10, pp. 47-60, March 31, 2008 2008.

[145] Y. Guo, P. Corke, G. Poulton, T. Wark, G. Bishop-Hurley, and D. Swain, "Animal Behaviour Understanding using Wireless Sensor Networks," in *Proceedings 31st IEEE Conference on Local Computer Networks,* , Tampa, FL, USA 2006, pp. 607-614.

[146] F. A. Tøgersen, F. Skjøth, L. Munksgaard, and S. Højsgaard, "Wireless indoor tracking network based on Kalman filters with an application to monitoring dairy cows," *Computers and Electronics in Agriculture,* vol. 72, pp. 119-126, 2010.

[147] T. Teixeria, G. Dublon, and A. Savvides, "A survey of human-sensing: methods for detecting presence, count, location, track, and identity," 11 September 2010.

[148] A.-J. Garcia-Sanchez, F. Garcia-Sanchez, F. Losilla, P. Kulakowski, J. Garcia-Haro, A. Rodríguez, *et al.*, "Wireless Sensor Network Deployment for Monitoring Wildlife Passages," *Sensors,* vol. 10, pp. 7236-7262, 2010.

[149] C. BenAbdelkader, R. G. Cutler, and L. S. Davis, "Gait Recognition Using Image Self-Similarity," *EURASIP Journal on Applied Signal Processing,* vol. 2004, pp. 572-585, 2004.

[150] S. L. Hannuna, N. W. Campbell, and D. P. Gibson, "Identifying quadruped gait in wildlife video," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, 2005, pp. I-713-16.

[151] Z. Liu and S. Sarkar, "Simplest Representation yet for Gait Recognition: Averaged Silhouette," presented at the 17th International Conference on Pattern Recognition, Cambridge, UK, 2004.

[152] L. Zongyi, L. Malave, and S. Sarkar, "Studies on Silhouette Quality and Gait Recognition," in *2004 IEEE Conference on Computer Vision and Pattern Recognition* Washington, DC, USA, 2004.

[153] T. Fei, A. K. Skidmore, V. Venus, T. Wang, M. Schlerf, B. Toxopeus, *et al.*, "A body temperature model for lizards as estimated from the thermal environment," *Journal of Thermal Biology,* vol. 37, pp. 56-64, 2012.

[154] J. Verner, R. N. Lehman, P. S. Forest, and R. E. Station, *Identifying individual bald eagles with voiceprints: a feasibility study*: U.S. Dept. of Agriculture, Forest Service, Pacific Southwest Forest and Range Experiment Station, 1982.

[155] W. Hu, T. Van Nghia, N. Bulusu, C. T. Chou, S. Jha, and A. Taylor, "The Design and Evaluation of a Hybrid Sensor Network for Cane-Toad Monitoring," in *Fourth International Symposium on Information Processing in Sensor Networks (IPSN 2005)*, Los Angeles, CA, USA,, 2005, pp. 503-508.

[156] P. H. Becker and H. Wendeln, "A new application for transponders in population ecology of the common tern," *The Condor,* vol. 99, pp. 534-538, 1997.

[157] D. Pavón, R. Limiñana, V. Urios, A. Izquierdo, B. Yáñez, M. Ferrer, *et al.*, "Autumn migration of juvenile short-toed eagles circaetus gallicus from southeastern Spain," *Ardea,* vol. 98, pp. 113-117, 2010/03/01 2010.

[158] S. Van der Spek, J. Van Schaick, P. De Bois, and R. De Haan, "Sensing human activity: GPS tracking," *Sensors,* vol. 9, pp. 3033-3055, 2009.

[159] M. Baratchi, N. Meratnia, and P. J. M. Havinga, "Recognition of periodic behavioral patterns from streaming mobility data," in *proceedings of the 10th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, Mobiquitous 2013*, Tokyo, Japan, 2013.

[160] M. J. Wisdom, N. J. Cimon, B. K. Johnson, E. O. Garton, and J. W. Thomas, "Spatial partitioning by mule deer and elk in relation to traffic," *Transactions of the 69th North American Wildlife and Natural Resources Conference,* pp. 509-530, 2004.

[161] S. Monroe, "Major and minor life events as predictors of psychological distress: Further issues and findings," *Journal of Behavioral Medicine,* vol. 6, pp. 189-205, 1983/06/01 1983.

[162] Z. Li, B. Ding, J. Han, R. Kays, and P. Nye, "Mining periodic behaviors for moving objects," presented at the Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, Washington, DC, USA, 2010.

[163] F. Verhein and S. Chawla, "Mining Spatio-temporal Association Rules, Sources, Sinks, Stationary Regions and Thoroughfares in Object Mobility Databases," in *Database Systems for Advanced Applications*. vol. 3882, M. Lee, K.-L. Tan, and V. Wuwongse, Eds., ed: Springer Berlin Heidelberg, 2006, pp. 187-201.

[164] F. Giannotti, M. Nanni, F. Pinelli, and D. Pedreschi, "Trajectory pattern mining," presented at the Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, San Jose, California, USA, 2007.

[165] L.-Y. Wei, Y. Zheng, and W.-C. Peng, "Constructing popular routes from uncertain trajectories," presented at the Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, Beijing, China, 2012.

[166] N. Mamoulis, H. Cao, G. Kollios, M. Hadjieleftheriou, Y. Tao, and D. W. Cheung, "Mining, indexing, and querying historical spatiotemporal data," presented at the Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, Seattle, WA, USA, 2004.

[167] M. Baratchi, N. Meratnia, and P. J. M. Havinga, "Finding frequently visited paths: dealing with the uncertainty of spatio-temporal mobility data," in *proceedings of the 2013 IEEE Eighth International Conference on Intellifent Sensor, Sensor Networks and Information Processing (ISSNIP'13)*, Melbourne, Australia, 2013, pp. 479-484.

[168] Z. Li, J. Wang, and J. Han, "Mining event periodicity from incomplete observations," presented at the Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, Beijing, China, 2012.

[169] A. Sadilek and J. Krumm, "Far Out: predicting long-term human mobility," in *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012, pp. 814-820.

[170] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," presented at the Proceedings of the 1993 ACM SIGMOD international conference on Management of data, Washington, D.C., USA, 1993.

[171] M. G. Elfeky, W. G. Aref, and A. K. Elmagarmid, "Periodicity detection in time series databases," *Knowledge and Data Engineering, IEEE Transactions on,* vol. 17, pp. 875-887, 2005.

[172] Y. Jiong, W. Wei, and P. S. Yu, "Mining asynchronous periodic patterns in time series data," *Knowledge and Data Engineering, IEEE Transactions on,* vol. 15, pp. 613-628, 2003.

[173] R. Yang, W. Wang, and P. S. Yu, "InfoMiner+: mining partial periodic patterns with gap penalties," in *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, 2002, pp. 725-728.

[174] S. Bar-David, I. Bar-David, P. C. Cross, S. J. Ryan, C. U. Knechtel, and W. M. Getz, "Methods for assessing movement path recursion with application to African buffalo in South Africa," *Ecology,* vol. 90, pp. 2467-2479, 2009.

[175] L. Riotte-Lambert, S. Benhamou, and S. Chamaillé-Jammes, "Periodicity analysis of movement recursions," *Journal of Theoretical Biology,* vol. 317, pp. 238-243, 1/21/ 2013.

[176] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing*. Upper Saddler River, NJ: Prentice Hall, 1999.

[177] Wikipedia. (2014). *Bernoulli distribution*. Available: http://en.wikipedia.org/wiki/Bernoulli_distribution

[178] M. Baratchi, N. Meratnia, P. J. M. Havinga, A. K. Skidmore, and A. G. Toxopeus, "A hierarchical hidden semi-Markov model for modeling mobility data," in *proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (Ubicomp'14)*, Seattle, Washington, 2014, pp. 401-412.

[179] A. Asahara, K. Maruyama, and R. Shibasaki, "A mixed autoregressive hidden-markov-chain model applied to people's movements," presented at the Proceedings of the 20th International Conference on Advances in Geographic Information Systems, Redondo Beach, California, 2012.

[180] W. Mathew, R. Raposo, and B. Martins, "Predicting future locations with hidden Markov models," presented at the Proceedings of the 2012 ACM Conference on Ubiquitous Computing, Pittsburgh, Pennsylvania, 2012.

[181] J.-G. Lee, J. Han, and K.-Y. Whang, "Trajectory clustering: a partition-and-group framework," presented at the Proceedings of the 2007 ACM SIGMOD international conference on Management of data, Beijing, China, 2007.

[182] L. Chen and E. al., "Robust and fast similarity search for moving object trajectories," presented at the In Proc. 2005 ACM SIGMOD Baltimore, Maryland, 2005.

[183] Y. Byoung-Kee, H. V. Jagadish, and C. Faloutsos, "Efficient retrieval of similar time sequences under time warping," in *In Proc. 14th Int. Conf. Data Engineering*, 1998, pp. 201-208.

[184] V. Bogorny, B. Kuijpers, and L. O. Alvares, "ST - DMQL: A Semantic Trajectory Data Mining Query Language," *International Journal of Geographical Information Science,* vol. 23, pp. 1245-1276, 2009/10/01 2009.

[185] S.-Z. Yu and H. Kobayashi, "A hidden semi-Markov model with missing data and multiple observation sequences for mobility tracking," *Signal Processing,* vol. 83, pp. 235-250, 2003.

[186] T. M. T. Do and D. Gatica-Perez, "Contextual conditional models for smartphone-based human mobility prediction," presented at the Proceedings of the 2012 ACM Conference on Ubiquitous Computing, Pittsburgh, Pennsylvania, 2012.

[187] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos, "Fast subsequence matching in time-series databases," presented at the In Proc. 1994 ACM SIGMOD Minneapolis, United States, 1994.

[188] M. Vlachos, G. Kollios, and D. Gunopulos, "Discovering similar multidimensional trajectories," in *Data Engineering, Proc.. 18th International Conference on*, 2002, pp. 673-684.

[189] L. Chen and R. Ng, "On the marriage of Lp-norms and edit distance," presented at the In Proc. Thirtieth int. conf. on Very large data bases Toronto, Canada, 2004.

[190] C.-C. Hung, W.-C. Peng, and W.-C. Lee, "Clustering and aggregating clues of trajectories for mining trajectory patterns and routes," *The VLDB Journal,* pp. 1-24, 2011/11/01 2011.

[191] G. Trajcevski and E. al., "Managing uncertainty in moving objects databases," *ACM Trans. Database Syst.,* vol. 29, pp. 463-507, 2004.

[192] F. Giannotti, M. Nanni, F. Pinelli, and D. Pedreschi, "Trajectory pattern mining," presented at the In Proc. 13th ACM SIGKDD, San Jose, California, USA, 2007.

[193] N. Pelekis and E. al., "Clustering uncertain trajectories," *Knowledge and Information Systems,* vol. 28, pp. 117-147, 2011.

[194] R. Montoliu, J. Blom, and D. Gatica-Perez, "Discovering places of interest in everyday life from smartphone data," *Multimedia tools and applications,* vol. 62, pp. 179-207, 2013.

[195] K. Farrahi and D. Gatica-Perez, "Extracting Mobile Behavioral Patterns with the Distant N-Gram Topic Model," in *Wearable Computers (ISWC), 2012 16th International Symposium on*, 2012, pp. 1-8.

[196] K. Farrahi and D. Gatica-Perez, "Probabilistic Mining of Socio-Geographic Routines From Mobile Phone Data," *Selected Topics in Signal Processing, IEEE Journal of,* vol. 4, pp. 746-755, 2010.

[197] T. M. T. Do, O. Dousse, M. Miettinen, and D. Gatica-Perez, "A probabilistic kernel method for human mobility prediction with smartphones," *Pervasive and Mobile Computing.*

[198] L. Song, D. Kotz, R. Jain, and X. He, "Evaluating location predictors with extensive Wi-Fi mobility data," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, 2004, pp. 1414-1424 vol.2.

[199] A. Asahara, K. Maruyama, A. Sato, and K. Seto, "Pedestrian-movement prediction based on mixed Markov-chain model," presented at the Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Chicago, Illinois, 2011.

[200] L.-Y. Wei, Y. Zheng, and W.-C. Peng, "Constructing popular routes from uncertain trajectories," presented at the In Proc. 18th ACM SIGKDD Beijing, China, 2012.

[201] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*: Elsevier, 2006.

[202] A. T. Palma, V. Bogorny, B. Kuijpers, and L. O. Alvares, "A clustering-based approach for discovering interesting places in trajectories," presented at the Proceedings of the 2008 ACM symposium on Applied computing, Fortaleza, Ceara, Brazil, 2008.

[203] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," *Ann. Math. Statist.,* vol. 37, pp. 1554-1563, 1966.

[204] S.-Z. Yu and H. Kobayashi, "An efficient forward-backward algorithm for an explicit-duration hidden Markov model," *Signal Processing Letters, IEEE,* vol. 10, pp. 11-14, 2003.

[205] H. Gao, J. Tang, and H. Liu, "Mobile Location Prediction in Spatio-Temporal Context," in *Mobile Data Challenge 2012* Newcastle, UK, 2012.

[206] M. Baratchi, N. Meratnia, and P. J. M. Havinga, "On the use of mobility data for discovery and description of social ties," in *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM'13)*, Niagara Falls, Canada, 2013, pp. 1229-1236.

[207] M. Granovetter, "The Impact of Social Structure on Economic Outcomes," *Journal of Economic Perspective,* vol. 19, p. 17, 2005.

[208] D. Cai, Z. Shao, X. He, X. Yan, and J. Han, "Community Mining from Multi-relational Networks," in *Knowledge Discovery in Databases: PKDD 2005*. vol. 3721, A. Jorge, L. Torgo, P. Brazdil, R. Camacho, and J. Gama, Eds., ed: Springer Berlin Heidelberg, 2005, pp. 445-452.

[209] M. K. Marsh, S. R. McLeod, M. R. Hutchings, and P. C. L. White, "Use of proximity loggers and network analysis to quantify social interactions in free-ranging wild rabbit populations," *Wildlife Research,* vol. 38, pp. 1-12, 2011.

[210] D. Mok, B. Wellman, and J. Carrasco, "Does Distance Matter in the Age of the Internet?," *Urban Studies,* vol. 47, pp. 2747-2783, November 1, 2010 2010.

[211] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: user movement in location-based social networks," presented at the Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, San Diego, California, USA, 2011.

[212] H. Kashima and N. Abe, "A Parameterized Probabilistic Model of Network Evolution for Supervised Link Prediction," in *Data Mining, 2006. ICDM '06. Sixth International Conference on*, 2006, pp. 340-349.

[213] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," presented at the Proceedings of the twelfth international conference on Information and knowledge management, New Orleans, LA, USA, 2003.

[214] S. Scellato, A. Noulas, and C. Mascolo, "Exploiting place features in link prediction on location-based social networks," presented at the Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, San Diego, California, USA, 2011.

[215] L. M. Aiello, A. Barrat, R. Schifanella, C. Cattuto, B. Markines, and F. Menczer, "Friendship prediction and homophily in social media," *ACM Trans. Web,* vol. 6, pp. 1-33, 2012.

[216] E. Gilbert and K. Karahalios, "Predicting tie strength with social media," presented at the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Boston, MA, USA, 2009.

[217] J. Zhuang, T. Mei, S. C. H. Hoi, X.-S. Hua, and S. Li, "Modeling social strength in social media community via kernel-based learning," presented at the Proceedings of the 19th ACM international conference on Multimedia, Scottsdale, Arizona, USA, 2011.

[218] L. Backstrom and J. Leskovec, "Supervised random walks: predicting and recommending links in social networks," presented at the Proceedings of the fourth ACM international conference on Web search and data mining, Hong Kong, China, 2011.

[219] J. Tang, T. Lou, and J. Kleinberg, "Inferring social ties across heterogenous networks," presented at the Proceedings of the fifth ACM international conference on Web search and data mining, Seattle, Washington, USA, 2012.

[220] X. Xiao, Y. Zheng, Q. Luo, and X. Xie, "Inferring social ties between users with human location history," *Journal of Ambient Intelligence and Humanized Computing,* pp. 1-17, 2012/12/01 2012.

[221] N. Eagle, A. Pentland, and D. Lazer, "Inferring friendship network structure by using mobile phone data," *Proceedings of the National Academy of Sciences,* vol. 106, pp. 15274-15278, September 8, 2009 2009.

[222] D. Wang, D. Pedreschi, C. Song, F. Giannotti, and A.-L. Barabasi, "Human mobility, social ties, and link prediction," presented at the Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, San Diego, California, USA, 2011.

[223] W. Dong, "Mutual information: inferring tie strength and proximity in bipartite social network data with non-metric associations," Master of science, University of Illinois at Urbana-Champaign, 2011.

[224] M. D. Domenico, A. Lima, and M. Musolesi, "Interdependence and predictability of human mobility and social interactions," presented at the Mobile Data Challenge 2012 (by Nokia) Workshop, 2012.

[225] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual Review of Sociology,* vol. 27, pp. 415-444, 2001.

[226] T. M. Cover and A. J. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[227] T. O. Kvalseth, "Entropy and correlation: Some comments," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 17

, pp. 217-219, 1987.

[228] I. Grosse, H. Herzel, S. V. Buldyrev, and H. E. Stanley, "Species independence of mutual information in coding and noncoding DNA," *Physical Review E,* vol. 61, pp. 5624-5629, 2000.

[229] Wikipedia. (22 Oct.). *Pearson product-moment correlation coefficient*. Available: http://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient

[230] A. T. Palma and E. al., "A clustering-based approach for discovering interesting places in trajectories," presented at the In Proc. 2008 ACM symposium on Applied computing, Fortaleza, Ceara, Brazil, 2008.

[231] Z. Xinxin, L. Lingjun, and X. Guoliang, "Checking in without worries: Location privacy in location based social networks," in *INFOCOM, 2013 Proceedings IEEE*, 2013, pp. 3003-3011.

[232] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility patterns," *Nature,* vol. 453, pp. 779-782, 06/05/print 2008.

[233] N. Meratnia and R. de By, "Spatiotemporal Compression Techniques for Moving Point Objects," in *Advances in Database Technology - EDBT 2004*. vol. 2992, E. Bertino, S. Christodoulakis, D. Plexousakis, V. Christophides, M. Koubarakis, K. Böhm*, et al.*, Eds., ed: Springer Berlin Heidelberg, 2004, pp. 765-782.

[234] J. Muckell, P. Olsen, Jr., J.-H. Hwang, C. Lawson, and S. S. Ravi, "Compression of trajectory data: a comprehensive evaluation and new approach," *GeoInformatica,* vol. 18, pp. 435-460, 2014/07/01 2014.

[235] B. Buchli, F. Sutton, and J. Beutel, "GPS-Equipped Wireless Sensor Network Node for High-Accuracy Positioning Applications," in *Wireless Sensor Networks*. vol. 7158, G. Picco and W. Heinzelman, Eds., ed: Springer Berlin Heidelberg, 2012, pp. 179-195.

[236] D. H. P. T. K. Douglas, "Algorithms for the reduction of the number of points required to represent a line or its caricature.," *Can Cartogr,* vol. 12, p. 10, 1973.

[237] M. Godau, "A natural metric for curves — Computing the distance for polygonal chains and approximation algorithms," in *STACS 91*. vol. 480, C. Choffrut and M. Jantzen, Eds., ed: Springer Berlin Heidelberg, 1991, pp. 127-136.

[238] R. Rana, M. Yang, T. Wark, C. T. Chou, and W. Hu, "A Deterministic Construction of Projection matrix for Adaptive Trajectory Compression," *arXiv preprint arXiv:1307.6923,* 2013.

[239] E. J. Keogh and M. J. Pazzani, "An Enhanced Representation of Time Series Which Allows Fast and Accurate Classification, Clustering and Relevance Feedback," in *KDD*, 1998, pp. 239-243.

[240] J. Muckell, J.-H. Hwang, V. Patil, C. T. Lawson, F. Ping, and S. S. Ravi, "SQUISH: an online approach for GPS trajectory compression," presented at the Proceedings of the 2nd International Conference on Computing for Geospatial Research & Applications, Washington, DC, USA, 2011.

[241] D. Feldman, A. Sugaya, and D. Rus, "An effective coreset compression algorithm for large scale sensor networks," presented at the Proceedings of the 11th international conference on Information Processing in Sensor Networks, Beijing, China, 2012.

[242] P. Agarwal, C. Procopiuc, and K. Varadarajan, "Approximation Algorithms for k-Line Center," in *Algorithms — ESA 2002*. vol. 2461, R. Möhring and R. Raman, Eds., ed: Springer Berlin Heidelberg, 2002, pp. 54-63.

[243] M. A. Abam, M. de Berg, P. Hachenberger, and A. Zarei, "Streaming Algorithms for Line Simplification," *Discrete & Computational Geometry,* vol. 43, pp. 497-515, 2010/04/01 2010.

[244] Z. Yan, N. Giatrakos, V. Katsikaros, N. Pelekis, and Y. Theodoridis, "SeTraStream: semantic-aware trajectory construction over streaming movement data," presented at the Proceedings of the 12th international conference on Advances in spatial and temporal databases, Minneapolis, MN, 2011.

[245] M. Potamias, K. Patroumpas, and T. Sellis, "Sampling Trajectory Streams with Spatiotemporal Criteria," in *Scientific and Statistical Database Management, 2006. 18th International Conference on*, 2006, pp. 275-284.

[246] R. Song, W. Sun, B. Zheng, and Y. Zheng, "PRESS: A novel framework of trajectory compression in road networks," *arXiv preprint arXiv:1402.1546,* 2014.

[247] G. Kellaris, N. Pelekis, and Y. Theodoridis, "Map-matched trajectory compression," *Journal of Systems and Software,* vol. 86, pp. 1566-1579, 6// 2013.

[248] H. Cao and O. Wolfson, "Nonmaterialized motion information in transport networks," in *Database Theory-ICDT 2005*, ed: Springer, 2005, pp. 173-188.

[249] Wikipedia. (2014, 22 Nov). *Forward Algorithm*. Available: http://en.wikipedia.org/wiki/Forward_algorithm

[250] Wikipedia. (19 Oct). *Markov property*. Available: http://en.wikipedia.org/wiki/Markov_property

[251] K. Chakrabarti, M. N. Garofalakis, R. Rastogi, and K. Shim, "Approximate query processing using wavelets," ed: Google Patents, 2004.