Spatiotemporal Data Mining, Analysis, and Visualization of Human Activity Data

by

Xun Li

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved November 2012 by the
Graduate Supervisory Committee:

Luc Anselin, Chair
Julia Koschinsky
Ross Maciejewski
Sergio Rey
William Griffin

ARIZONA STATE UNIVERSITY

December 2012

ABSTRACT

This dissertation addresses the research challenge of developing efficient new methods for discovering useful patterns and knowledge in large volumes of electronically collected spatiotemporal activity data. I propose to analyze three types of such spatiotemporal activity data in a methodological framework that integrates spatial analysis, data mining, machine learning, and geovisualization techniques. Three different types of spatiotemporal activity data were collected through different data collection approaches: (1) crowd sourced geo-tagged digital photos, representing people's travel activity, were retrieved from the website Panoramio.com through information retrieval techniques; (2) the same techniques were used to crawl crowd sourced GPS trajectory data and related metadata of their daily activities from the website OpenStreetMap.org; and finally (3) preschool children's daily activities and interactions tagged with time and geographical location were collected with a novel TabletPC-based behavioral coding system. The proposed methodology is applied to these data to (1) automatically recommend optimal multi-day and multi-stay travel itineraries for travelers based on discovered attractions from geo-tagged photos, (2) automatically detect movement types of unknown moving objects from GPS trajectories, and (3) explore dynamic social and socio-spatial patterns of preschool children's behavior from both geographic and social perspectives.

TABLE OF CONTENTS

Page

LIST OF TABLES vi

LIST OF FIGURES... viii

INTRODUCTION.. 1

References...7

ESSAY 1: Building an Intelligent Tourist Trip Plan System based

on Geo-tagged Photos 8

Abstract...8

1.1 Introduction 10

1.2 Related Work...14

1.2.1 Exploring travel attractions with geo-tagged photos 14

1.2.2 Mining travel patterns from geo-tagged photos.......................17

1.2.3 Recommending travel destinations 20

1.2.4 Recommending travel itineraries 22

1.3 Methodology 26

1.3.1 Finding POIs from Geo-tagged Photos 27

1.3.2 Properties of POI 30

1.3.3 Traveling Graph Model 31

1.3.4 Travel Patterns 34

1.3.5 Tourist Trip Design 35

1.3.5.1 Problem Definition 38

1.3.5.2 Heuristic Solution 40

1.4 Experiments 45

1.4.1 Data 45

1.4.2 POIs based on geo-tagged photos 46

1.4.3 Travel patterns based on geo-tagged photos 51

1.4.4 Travel Path Plan 52

1.5 Conclusion 59

1.6 References 63

APPENDIX A

ALGORITHMS 69

ESSAY 2: Introducing Complexity Measures to Trajectory Classification Modeling to Automatically Detect Movement Types with Unknown GPS Trajectories 74

Abstract...74

2.1 Introduction 76

2.2 Related Work...81

2.2.1 Trajectory Classification based on Geometric Shape 83

2.2.2 Trajectory Classification based on Movement Parameters 85

2.3 Complexity of Movement 88

2.3.1 Geometric Complexity of Movement and Fractal Dimension 89

Page

2.3.2 Complexity of Movement Parameters and

Approximate Entropy 91

2.4 Methods 96

2.4.1 Data Preprocessing 97

2.4.2 Feature Extraction and selection 98

2.4.3 Classification Model 101

2.4.4 Trajectory Classification, Prediction and Evaluation 103

2.5 Experiments and Results 105

2.5.1 Data Collection 105

2.5.2 Data Preprocessing, Feature Extraction and Selection 106

2.5.3 Experiments..110

2.5.4 Results 113

2.6 Conclusion 117

2.7 References 119

ESSAY 3: Using ESDA with Social Weights to Analyze Spatial and

Social Patterns of Preschool Children's Behavior..123

Abstract..123

3.1 Introduction 124

3.2 Related Work..127

3.3 Methods 132

Page

3.3.1 Spatial Autocorrelation and Local Indicators

of Spatial Association 133

3.3.2 Global and Local Spatial Autocorrelation

with Social Weights 136

3.3.2.1 Social Distance and Social Networks 138

3.3.2.2 Spatial Autocorrelation with Social Weights 141

3.3.2.3 LISA with Social Weights 143

3.4 Data, Experiments and Results 145

3.4.1 Data 145

3.4.2 Case Study Applications 149

3.4.3 Results 151

3.4.3.1 Spatial Patterns of Preschool

Children's Behavior 151

3.4.3.2 Spatial Autocorrelation of Preschool

Children's Activities 160

3.5 Conclusion 168

3.6 References 171

APPENDIX B

ADDITIONAL GLOBAL MORAN'S I TEST RESULTS 175

CONCLUSION 178

LIST OF TABLES

Table Page

1.1 Algorithm: A k-d Tree based OPTICS Algorithm for Massive Dataset............70

1.2 Algorithm: Extract hierarchical clustering structure from OPTICS reachability plot..70

1.3 Algorithm: Mining Classic ROI based Travel Routes 72

1.4 Algorithm: Iterated Local Search heuristic algorithm for multi-stay TOPTW problem 73

2.1 Descriptive Statistics and Structural Complexity Measures of 4 Movement Parameters (a,b,c,d) for 4 Randomly Selected Trajectories.......................107

2.2 Geometric Complexity Measures (Fractal Dimensions) of 4 Randomly Selected Trajectories with 4 Different Movement Types 109

2.3 Correlation Coefficients between Movement Parameters of 4 Movement Types 110

2.4 Cross-model Comparisons of Classification Models Using Experiment 1 Dataset: 320 Training Data/80 Testing Data.....................112

2.5 Confusion Matrix of Accuracy for 4-class Trajectory Classification Problem in Experiment 1 114

2.6 Confusion Matrix of Accuracy for 4-class Trajectory Classification Problem in Experiment 2 115

2.7 Comparison of Classification Results of Experiment 1 and 2 116

3.1 15 Observed Activities with Brief Definitions 147

Table Page

3.2 Testing Results for Global Spatial Autocorrelation of Children's Spatial Behaviors using the Moran's *I* Statistic (Queen Weights Matrix)...........176

3.3 Testing Results for Global Spatial Autocorrelation of Boy's and Girl's Spatial Behaviors using the Moran's *I* Statistic (Queen Weights Matrix)...........177

3.4 Testing Results for Global Spatial Autocorrelation with Social Weights using the Moran's I Statistic (Classroom 1) 164

3.5 Testing Results for Global Spatial Autocorrelation with Social Weights using the Moran's I Statistic (Classroom 2) 165

LIST OF FIGURES

Figure Page

1 Proposed methodological framework for spatiotemporal data mining, analysis, and visualization of human activity data 4

1.1 The process model of discovering popular places and movement patterns from geo-tagged photos 27

1.2 An illustration of the construct step in the proposed heuristic algorithm for multi-day and multi-stay travel trip plan 41

1.3 Geographical distribution of 118,736 geo-tagged photos that used in this case study of Australia 46

1.4 Reachability plot of OPTICS clustering results on geo-tagged photos. The red line is a demonstration for finding DBSCAN clusters at different hierarchies 48

1.5 The dendrogram that was generated based on the OPTICS reachability plot with real geo-tagged photos.. 49

1.6 POIs, ROIs and travel patterns (yellow arrow flows) that are mined from geo-tagged photos and displayed at different spatial scales in Australia: (A) country (B) province (c) city.. 51

1.7 A 2-day tourist trip itinerary, which starts and ends at Sydney International Airport 56

1.8 A 4-day tourist trip itinerary, which starts and ends at Sydney International Airport 58

Figure Page

2.1 4 different randomly selected GPS trajectories (car, bike, run, walk)

in 2D (upper) and in 3D (lower, with vertical axis representing time)..... 90

2.2 Plots of velocity against time for: car, bike, run, and walk

(from top to bottom) of 4 randomly selected trajectories with

4 different movement types 94

2.3 Illustration of computing general movement parameters such as:

moving speed, turning angle, displacement etc.. 99

2.4 Receiver operating characteristic (ROC) and area under ROC curves

comparison of "one-against-rest" binary classification results of

Bike, Walk, Run and Car in experiment 1 and 2...................................... 116

3.1 Process diagram to illustrate global and local spatial autocorrelation with

spatial weights as applied in this essay 137

3.2 Process diagram to illustrate the application of global and local spatial

autocorrelation with social weights in this essay 140

3.3 A map of the preschool structure 148

3.4 A TabletPC based behavioral coding system: Coders use a digital

pen to record preschool children's behavior in a GIS-based

graphic user interface 148

3.5 Kernel density maps of all observed indoor social activity for

semester 2008 Fall (Upper), and 2009 Spring (Lower) 149

3.6 The plots of Global Moran's I Z values of four behaviors in two semesters

from observations of mixed gender, boy only and girl only. 155

Figure Page

3.7 LISA maps by type of behavior and date, for all children in semester 2008 Fall and 2009 Spring (only cluster cores are shown)............................... 157

3.8 LISA maps by type of behavior and date, for boys only in semester 2008 Fall and 2009 Spring (only cluster cores are shown)............................... 158

3.9 LISA maps by type of behavior and date, for girls only in semester 2008 Fall and 2009 Spring (only cluster cores are shown)............................... 159

3.10 Top 3 local social network autocorrelation graphs by type of activities in classroom 1 in 2008 Fall and 2009 Spring semesters............................... 166

3.11 Top 3 local social network autocorrelation graphs by type of activities in classroom 2 in 2008 Fall and 2009 Spring semesters............................. 167

Introduction

This dissertation contributes to addressing the research challenge of discovering useful patterns and knowledge in increasingly ubiquitous, large-scale, electronically collected, spatiotemporal activity data. Digital spatiotemporal data of human activities, much of it accessible online, has been increasing exponentially. For instance, the technologies of location-aware devices, such as GPS and WiFi, are becoming more pervasive. Such devices are now prevalent in vehicles, cellular phones and personal mobile devices and can track and record people's location or record locations of events (e.g. crime).

Meanwhile, with the development of Web 2.0, which focuses on user-centered design for interactive information sharing, many people are sharing their geolocation movement traces on public websites. This makes this type of directly collected data widely accessible (Haklay and Weber 2008). In indoor environments, where GPS systems do not work, radio frequency identification (RFID) and Bluetooth can be used to position people's geographic location. Further, as shown in chapter 3 in this dissertation, TabletPCs can be configured to track children's movement in a preschool (indoor and outdoor) by coders using digital pens.

Beyond these direct methods of collecting spatiotemporal activity data, these data can also be retrieved indirectly from location proxies. For example, people's space-time consumption behavior can be extracted based on their ATM transactions as "location proxies". Digital photos that are images of places people visited can also serve as "location proxies" to represent their space-time tourism behavior if listed in chronological order. Thanks to Web 2.0, the large numbers of geo-tagged photo data are also publicly available for researchers (Goodchild 2007).

Such indirectly collected spatiotemporal data are often an abstract representation of people's actual movement in physical space. These enormous amounts of directly and indirectly collected types of spatiotemporal data of human activities become a new data source for studying human socio-spatial behavior. How to collect these data and efficiently and accurately discover useful social and geographic patterns and knowledge from these data is the research question of this dissertation.

The three research essays in this dissertation address the following gaps in existing data mining research:

(1) To make travel recommendations for travelers based on discovered popular places and travel patterns from large numbers of online geo-tagged photographs, existing research has applied spatial analysis, data mining, machine learning and dynamic programming techniques to address the research challenges. However, several research gaps remain in this context that still need to be addressed: The first gap is that, due to the additional complexity involved, little existing research focuses on real-life travel itinerary planning problems, such as making multi-day and multi-stay (different accommodations during travel) travel itineraries for inexperienced tourists with limited time. The second gap is that the discovered travel knowledge and patterns, such as attractiveness scores, average

visiting times of points-of-interest (POIs) and travel reoccurrences between POIs, are usually isolated from tackling travel itinerary planning problem. Such data mined travel knowledge and patterns contain important information for making quality travel itineraries. However, no current research takes advantage of this information to build an intelligent travel itinerary planning system.

(2) To train a trajectory classification model from trajectories and automatically detect movement types of unknown trajectories, existing research classifies trajectories based on classic geometric shape-based approaches by utilizing the geometric characteristics of movement. However, a major limitation of this approach is that it restricts trajectory comparisons to the same geographic region since all predefined trajectory categories are tied to this region. Classification methods based on movement parameters can overcome this problem but the accuracy of classification depends heavily on selecting appropriate movement features from trajectories. Recent research attempts to extract local movement profiles to improve the accuracy of classification but is restricted to fixed-size trajectories.

(3) To find social and spatial patterns of preschool children's behavior from micro-social spatiotemporal activity data, existing studies on preschool behavior have applied GIS, statistical analysis, spatial analysis, and social network analysis techniques. However, most of these studies are restricted to either a social context or a spatial context. For instance, geographic data and methods are usually used for discovering spatial patterns (spatial heterogeneity, spatial externalities and spatial spillovers) while social data and methods are

mostly used for discovering social network patterns in preschool behavior. This research gap in simultaneously applying spatial and social methods to preschool children's behavior limits research on the joint relationship between geographic and social settings and the socialization of preschool children.

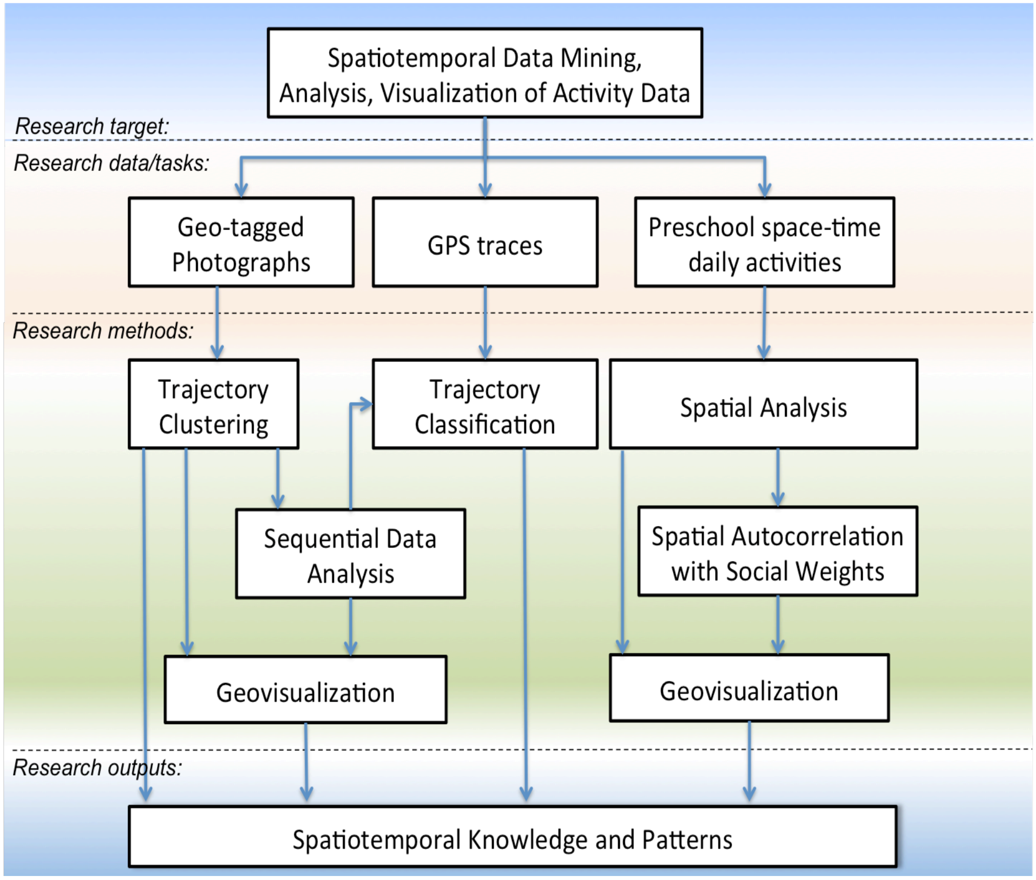

Spatiotemporal Data Mining, Analysis, Visualization of Activity Data
Research target:
Research data/tasks:
Geo-tagged Photographs
GPS traces
Preschool space-time daily activities
Research methods:
Trajectory Clustering
Trajectory Classification
Spatial Analysis
Sequential Data Analysis
Spatial Autocorrelation with Social Weights
Geovisualization
Geovisualization
Research outputs:
Spatiotemporal Knowledge and Patterns

Figure 1: Proposed methodological framework for spatiotemporal data mining, analysis, and visualization of human activity data.

To address these three research gaps, I propose to analyze three types of such spatiotemporal activity data in a methodological framework that integrates spatial analysis, data mining, machine learning, spatial optimization and geovisualization techniques (see figure 1). The main contributions of the three essays are as follows: In essay 1, a hierarchical region-of-interest (ROI) based graph model is first applied to automatically discover attraction places and related properties, and travel patterns from geo-tagged photos at different spatial scales. Then, this research proposes a modified Iterated Local Search (ILS) heuristic algorithm to efficiently find an approximate optimal solution for the multi-day and multi-stay travel itinerary recommendation problem by using additional travel information, such as the attractive score of a POI, reoccurrence weights of trips etc., which are mined from geo-tagged photos. Crowd sourced geo-tagged digital photos, representing people's travel activity, were retrieved from the website Panoramio.com through information retrieval techniques to test this model.

In essay 2, I propose a trajectory classification model based on movement parameters that I introduce, which are geometric complexity measures of trajectories and structural complexity measures of movement parameters. This model automatically detects movement types of unknown moving objects from GPS trajectories. The performance of this model is evaluated in an experiment that applies the same techniques as in 1) to crawl crowd sourced GPS trajectory data and related metadata of daily activities from the website OpenStreetMap.org.

In essay 3, exploratory spatial data analysis with social weights is applied to explore dynamic social and socio-spatial patterns of preschool children's behavior from both geographic and social perspectives. The data for this project were collected by a team of researchers as part of a larger NSF-funded project[1]: preschool children's daily activities and interactions, tagged with time and geographical location, were collected with a novel Tablet-PC based behavioral coding system that the author helped to implement.

The first essay is titled "Building an Intelligent Travel Trip Plan System based on Geo-tagged Photos", followed by an essay on "Introducing Complexity Measures to Trajectory Classification Modeling to Automatically Detect Different Movement Types with Unknown GPS Trajectories." The dissertation concludes with an essay on "Using ESDA with Social Weights to Analyze Spatial and Social Patterns of Preschool Children's Behavior."

[1] This work was funded by the National Science Foundation (NSF) as part of the project "Modeling time, space, and behavior: Combining ABM & GIS to create typologies of playgroup dynamics in preschool children" by William Griffin, Paul Torrens, Jennifer Fewell (2006-2011). The data were collected by Casey Sechler, Jillian Smith and other ASU undergraduate and graduate students who have participated in this project.

References

Goodchild, M. F. (2007) Citizens as sensors: the world of volunteered geography. GeoJournal*,* 69**,** 211-221.

Haklay, M. & P. Weber (2008) OpenStreetMap: User-generated street maps. Pervasive Computing*, IEEE,* 7**,** 12-18.

ESSAY 1

Building an Intelligent Tourist Trip Plan System based on Geo-tagged Photos

Abstract

By utilizing large amount of public available geo-tagged photos, existing research can successfully discover attractions places and travel patterns, and make simple travel recommendations for travelers. However, few of them focuses on complicated real-life travel plan problems, such as making multi-day and multi-stay (different places of accommodation) travel itinerary for inexperienced tourists with limited time budget. By integrating and extending the state-of-the-art techniques in spatiotemporal data mining and solutions of orienteering problems in operational research, this research develops a novel intelligent tourist trip plan system based on geo-tagged photos to automatically generate useful travel knowledge and make travel itinerates. Specifically, this system addresses an important challenge in existing research: to efficiently solve the tourist trip plan problem in a way that can benefit actual travel planning by leveraging massive geo-tagged photos.

First, the Order Points To Identify the Clustering Structure (OPTICS) clustering algorithm is applied to explore attractive places and points-of-interest (POIs) from geo-tagged photos. Properties of POI, such as attractive score and suggested visiting time, are also extracted from geo-tagged photos. Then, a traveling graph model is generated through reconstructed individual travel trips using geo-tagged photos. The reoccurrence weight of trips between POIs is

computed using a sequential pattern recognition algorithm. Further, travel patterns at different spatial scales are mined for travel references using this graph model and the hierarchical clustering results of OPTICS. Finally, I develop a modified Iterated Local Search (ILS) heuristic algorithm to efficiently find an approximate optimal solution to the multi-day and multi-stay tourist trip plan problem using discovered POIs and travelling graph model as additional travel information. To demonstrate the utility of this approach, I retrieve geo-tagged photos that were taken in Australia from the online photo sharing website Panoromia.com to develop optimal multi-day and multi-stay travel itineraries for tourists.

1.1 Introduction

Volunteered geographic information provides researchers with great opportunities and challenges to develop new methods for discovering underlying geographical knowledge of travel behavior (Goodchild 2007). Photos shared through websites are one example of volunteered geographical information. Several technological advances have enabled the widespread availability of electronic photos tagged with geographical location. For one, accurate location-aware chips are widely manufactured and embedded in digital cameras and smart phones with cameras. Further, location-aware techniques can detect accurate geographical location of cameras in both outdoor (e.g. GPS) and indoor (e.g. cellular network, WiFi, RFID) environments. With these technologies, geographical location and time information can be automatically written into digital photos when people take pictures. Meanwhile, with the rapid development of Web 2.0, more and more people are publishing or sharing their digital photos with friends via photo sharing websites. For example, at Panoramio.com and Flickr.com, several billion geo-tagged photos are publicly available for browsing and searching. For instance, I experimented with retrieving 36 million geo-tagged photos from Panoramio.com for this dissertation.

Photo-sharing websites generally provide an online mapping system for Internet users to interactively understand what a specific place in the world looks like by browsing the photos mapped at that place. Moreover, these geo-tagged photos not only contain visual information of places, but also provide rich spatial and temporal information of people’s travel behavior. In recent years, publicly available geo-tagged photos have been successfully used in studies of tourism, computer graphics, mobile research and geography. As discussed in more detail in the literature review in section 1.2, existing work mainly focuses on manipulating the spatial information of geo-tagged photos to discover landmarks, find attractive areas in a city or explore the borders of city center neighborhoods. Other studies simultaneously utilize the spatial, temporal and visual information of geo-tagged photos to mine travel patterns, automatically determine the location of un-tagged photos, reconstruct three-dimensional scenes of landmarks, and provide travel recommendation services.

For people who are going to visit a region or a city, such useful travel information and recommendations can help them to get a better sense of a place and how to plan their travel trip. However, there are often more attractions in a city than can be visited within limited travel time. To determine which places to go in what time during their available travel days, tourists need to incorporate several pieces of information simultaneously, such as the airport where they arrive and leave, the number of days available for travel, the attractiveness of travel destinations, the travel time between destinations or even the opening and closing times of destinations. Since this is complicated, inexperienced tourists often solve such problem by seeking guidance from travel guidebooks, traveling experts, such as travel agents or local residents, or finding answers from social media resources, such as Lonely Planet[2] or TripAdvisor[3]. However, it can take tourists a lot of time

[2] http://www.lonelyplanet.com
[3] http://www.tripadvisor.com

to process large amounts of travel information and translate this info a feasible travel plan.

To solve this travel trip-planning problem efficiently, research based on geo-tagged photos utilizes various approaches to develop intelligent travel trip planning systems for automatically recommending travel plans for travelers. When incorporating temporal information with geo-tagged photos, the spatiotemporal representation of photos actually provides a "digital footprint" that can be used to reconstruct travel trips traversed by photographers (Girardin et al. 2008). By utilizing these photo-based travel routes and experiences, current research tries to find classical travel routes, recommend best travel destinations and itineraries for travelers. A common feasible approach to build a travel planning system is to first discover travel attractions and related information, such as attractive score, average visiting hours etc., from geo-tagged photos, and then develop algorithms to find and recommend best travel routes for tourist based on their travel premise (e.g. start/end locations and travel time), Section 1.2.4 reviews the existing research in this research area.

However, a remaining challenge of this approach that needs to be addressed is related to not only solving the travel planning problem efficiently but also in a way that takes time and budget constraints travelers into account. Since travel recommendation research based on geo-tagged photos is still in the early stages, most research has focused on relatively simple travel planning problems that does not consider the available traveling time of tourists, the opening and closing times of attractions, or different places of accommodation for multi-day travel. Therefore, their applicability to real-life traveling situations is limited. Besides, when tackling more complex travel planning problems, slow performance becomes another challenge because exponentially increased complexity. Recent research on the Orienteering Problem (OP) provides efficient heuristic algorithms to tackle similar complex travel planning problems. In this case, attractions are usually provided by places like tourist offices and the attractive scores are predefined categorically according to the types of attractions. Borrowing such efficient algorithms to solve a travel-planning problem based on geo-tagged photos in a way that more realistically models with traveler's budget and time constraints remains a research gap. The orienteering problem and its extensions are discussed in section 1.3.5.

To address this challenge in this essay, I develop an intelligent tourist trip plan system based on geo-tagged photos that mines useful travel knowledge and finds optimal multi-day and multi-stay travel route plans. I achieve this through data mining and knowledge discovery while making use of spatial clustering, pattern recognition, meta-heuristic solutions for the orienteering problem, and geovisualization. Specifically, I use the ordering points to identify the clustering structure (OPTICS) algorithm (Ankerst et al. 1999) to explore the attractive places as POIs from geo-tagged photos. Then, I construct a traveling graph model to mine travel patterns from different spatial scales. With the discovered POIs and travel graph model, I develop a modified Iterated Local Search (ILS) heuristic algorithm to efficiently find and approximate optimal solution. The usefulness of this system is demonstrated in an application of Australia, using geo-tagged photos retrieved from the website Panoramio.com. The potential broader impacts of this work include the study of automatic and intelligent travel planning, tourism patterns and related economic activities as well as the provision of location-based services as travel guides, web recommendation system, or personal travel assistant for smart phone applications and other devices.

1.2. Related Work

The spatiotemporal information embedded in large amount of user generated geo-tagged photos implicitly provides rich travel-related information for tourists. In recent years, by leveraging publicly available online geo-tagged photos and related meta-data, much research tried to discover useful travel information (e.g. landmarks/attractions and their visiting time preferences, travel behaviors and travel patterns) and develop intelligent and expert system to recommend travel destinations and travel itineraries to potential users. In general, existing work has focused on three tasks: (1) exploring travel attractions from geo-tagged photos; (2) mining travel patterns from geo-tagged photos;(3) recommending travel destinations based on geo-tagged photos and (4) recommending travel itineraries based on geo-tagged photos. The method proposed in this essay builds on and advances this foundation to overcome some of the key remaining challenges in current research.

1.2.1 Exploring travel attractions with geo-tagged photos

People take photos at the places that attracted them (Zheng et al. 2011b). Travel attractions could be some specific geographical locations/areas that have prominent features, including cultural, natural (e.g., landmarks and endangered species) or geographic features. Recently, many studies have investigated landmarks or attractive regions from photos tagged with geographical locations. Despite using different types of case studies, most research applied similar methodologies: first, spatial clusters are discovered from geo-tagged photos using various spatial clustering algorithms; then, representative photos or locations are generated as landmarks for each spatial cluster. Regarding the types of clustering algorithms utilized, current studies can be classified into those applying the partitioning clustering approach; the density based clustering approach and the hierarchical clustering approach. These clustering methods and their applications in the context of this essay's research are discussed next.

In partitioning clustering approach, the *k*-means algorithm (MacQueen 1967) that takes the number of clusters as an input parameter, and uses the mean value of the points in a cluster as the centroid is popular in research based on geo-tagged photos. Ahern et al. (2007) applied *k*-means to explore geographical clusters of geo-tagged photos in equal-sized “tiles” at two different geographical scales. In Chen et al. (2009), *k*-means was used to find spatial clusters from geo-tagged photos in selected metropolitan areas. However, *k*-means is a fixed-cluster approach and the value of *k* is difficult to predefine. Besides, the clusters explored by *k*-means are spherical -- hence it is not suitable for finding arbitrary shaped clusters, which are common in natural environments.

The collective behavior of photographers is represented by an uneven geographic distribution of geo-tagged photos: more photos are taken at more popular places. Therefore, density-based approaches become natural solutions for finding POIs or ROIs from geo-tagged photos. For example, Hoashi et al. (2009) applied the DBSCAN algorithm to find clusters from 12,498 geo-tagged photos in the Tokyo metropolitan area. For each cluster, the centroid with the highest photo density is used to query a list of names from travelogues to describe this area. Kisilevich et al. (2010b) apply the same clustering method to find attractive regions from 20,200 geo-tagged photos in the city of Munich. Kisilevich and colleagues (2010c) further developed a modified DBSCAN algorithm, called P-DBSCAN, to explore attractive regions from 28,707 geo-tagged photos of 4,160 photographers in the Washington D.C. area. In their algorithm the density threshold *MinPts* is defined as the minimum number of photographers in the neighborhood, so that all points in the cluster have equal weight.

Another popular density based approach applied in geo-tagged photo research is Mean Shift clustering (Fukunaga and Hostetler 1975), which is a non-parametric technique that does not require prior knowledge of the shape or the number of clusters. It assumes that the distribution of points can be approximated via kernel density estimation, so that dense clusters can be found as the mode of the probability density functions. In Lu et al. (2010), Mean Shift clustering was successfully applied to automatically detect 300,000 clusters in the world from near 20 million geo-tagged photos. To avoid the number bias of the photos taken by different people, Crandall and colleagues (2009) divided the space into 1 degree and .001 degree buckets to represent metropolitan scale (100km) and landmark scale (100m). In each bucket, only one photo for each photographer is sampled. After mean shift clustering, the photos in the buckets that are located at the corresponding clustering centers are selected for finding statistically significant landmarks.

The hierarchical clustering algorithms decompose points into several levels of nested clusters. The result is usually organized as a tree where the root is the whole point set and each node is a sub point set. Zheng et al. (2009) used the agglomerative hierarchical clustering approach to find spatial clusters from geo-tagged photos. It is worth noting that the structure of hierarchical clustering results is useful for map-based browsing because it is convenient to organize and display POIs or ROIs at different zoom levels on a limited computer screen. In Jaffe et al. (2006), a modified Hungarian hierarchical clustering algorithm (Goldberger and Tassa 2008) is used to find spatial clusters from geo-tagged photos and organize them in a tree structure. For each node (sub-cluster) on a different hierarchical level, representative photos and textual tags are identified. Therefore, in their map-based system, the corresponding representative photos and tags can be displayed at different zoom levels.

1.2.2 Mining travel patterns from geo-tagged photos

The large online database of publicly shared geo-tagged photos also provides researchers with opportunities to discover individual and collective travel patterns, which are important for tourism research and the tourist industry. Individual geo-tagged photo collections represent a type of “digital footprint” (Girardin et al. 2008) that can reflect the spatial-temporal movement of tourists who take photos. Therefore, travel patterns can be revealed from the collective behavior of photographers. Several researchers (Clements et al. 2010, Gao et al. 2010, Cao et al. 2010) tried to develop various approaches to mine travelers' collective behavior or travel patterns from their geo-tagged photos and then making travel recommendations for Internet users based on the queried information.

Girardin et al. (2008) used geo-visualization techniques to statically and qualitatively analyze tourist patterns from geo-tagged photos taken by 753 travelers in Rome, Italy over a three-month period. In their approach, they manually explored spatial travel preference clusters from photos. Then, the travel patterns were generated and visualized as weighted desire lines by aggregating individual travel paths, which were captured as the sequential travel preference of users. An advanced geo-visualization technique is proposed by Jankowski et al. (2010) to discover movement patterns from geo-tagged photos. In their approach, the area is divided into Voronoi tessellations by using a density based convex shape clustering algorithm developed by Adrienko and Adrienko (2010). Then, the collection of photo sequences was converted to aggregate flows among compartments to Voronoi polygons. To explore the movement patterns, they used the flow mapping technique (Tobler 1987) to visualize aggregate flows at different time periods. The number of actual movements of photographers between two compartments defines the width of travel flow. They defined a popularity threshold to filter out travel flows with a number of actual movements that is lower than certain values as well as a distance threshold to analyze either long travel flows (>=3km) or short travel flows (<=1500m). In their following work (Andrienko et al. 2009), they applied density based clustering algorithm to find popular travel places based on geo-tagged photos from Panoramio.com, and built flow maps to show aggregated travel patterns between places.

Kisilevich et al. (2010a) proposed a novel approach to mine travel patterns from geo-tagged photos in two European cities. In their approach, each photo was assigned a textual tag named based on its nearby POI. For photos without nearby POI, they applied the DBSCAN algorithm to generate spatial clusters and manually assigned names to those photos. Then, individual travel routes were represented by photo tag sequences. To discover travel patterns from these travel routes, they used the Teiresias algorithm (Rigoutsos and Floratos 1998), a combinatorial pattern discovery algorithm for analyzing DNA sequences in bioinformatics used to discover recurrent maximal patterns within sequences. The results of frequently appearing travel routes were textually described as POI sequences.

In Zheng et al. (2011a), the authors first used DBSCAN to explore regions-of-attractions (ROAs) in order to mine travel pattern from geo-tagged photos in Paris and London. Travelers' spatiotemporal movements were represented as visit sequences of ROAs. Then, they applied the Markov chain model (Diaconis 2008) to analyze the transition probabilities of traveling between ROAs. The tourist traffic flow among different ROAs with higher transition probability was further explored and visualized on an online digital map.

1.2.3 Recommending travel destinations

The travel history that is embedded in travelers' photos, combined with the aggregated travel information in photos, enable researchers to recommend possible travel destinations for tourists. The basic idea is to find the most likely favored landmarks for a given user in a new place based on similar users with similar travel experiences in other places in the past. A variety of approaches have been developed to make recommendations of travel destinations for tourists from geo-tagged photos.

To recommend personalized landmarks in a target city to users, an intuitive approach that utilizes their geo-tagged photos in the Flickr community is proposed by Clements et al. (2010). For a specified user, their approach tries to find all other users who have similar travel habits in other attractive cities. Then, it summarizes their favorite landmarks in the target city as the best candidates for the specified user. The similarity between two users is defined based on the geographical distributions of their photos, and is computed as the sum over the minimum of two related Gaussian kernel convolutions, which are generated from peaks in the Mean Shift clustering of geo-tagged photos.

Shi et al. (2011) formulate the travel landmarks recommendation task as a collaborative filtering problem by using geo-tagged photos contributed by users to Flickr. The landmark for each photo is extracted from user tags and closest geo-tagged Wikipedia articles. They proposed a category-regularized matrix factorization approach to recommend landmarks to users based on user-landmark preference and landmark similarity. The user-landmark preference is defined as the number of users' photos of a landmark, and the similarity between landmarks is defined by category-based similarity, where categories of landmarks were extracted from Wikipedia.

Caludio et al. (2010) proposed an approach based on Random Walks with Restart (RWR) (Tong and Faloutsos 2006) to tackle the problem of predicting the probabilities of visiting other POIs given a user's visiting history extracted from geo-tagged photos. Top-k ranked POIs in the study area will be recommended to users. In their approach, RWR is based on a graph model, where nodes are POIs, edges are transitions and weights are conditional probabilities of any pair of POIs. This graph model is constructed using an accumulative visiting history of all users in Flickr. In their algorithm, the RWR works by sending a set of random walkers to the graph from POI based on the visiting history and collecting the conditional probability of reaching other POIs that were not part of the visiting history. Then, their algorithm selects the most relevant POIs with the highest probabilities for a particular user.

Besides considering the popularity of landmarks for recommendations, Van Canneyt et al. (2012) proposed a time-dependent probabilistic approach to take the time context into account when recommending landmarks to users. In their probabilistic model, the time context (e.g. time interval, day of week and month of year) is considered as a combined condition for estimating joint probabilities with POIs. By doing so, the same POI could get different recommendation probabilities given different visiting times (e.g. visiting a museum on Saturday might get a higher probability than Sunday if it is free on

Saturday). Therefore, their model can recommend different travel places to users given different expected travel time schedules.

To address the problem of recommending a package of travel landmarks given time and money cost budget constraints, Xie et al. (2011) utilize Yelp[4] and Wikipedia[5] datasets to develop a composite recommendation system for travel planning. The POI dataset is extracted from Yelp and the visiting value of POI is assigned as the ranking score. The money and time cost for each POI is estimated based on the information extracted from Yelp and Wikipedia. In their system, this composite recommendation problem is formulized and solved as a variation of the classical 0/1 2-dimensional knapsack problem (Kellerer et al. 2004). They designed their algorithm to run a knapsack solver iteratively by adding new POIs into an existing candidate list. This algorithm can find top-k POI sets, with the highest sum of visiting values while constraining the sum of time and money expenses under the given budget.

1.2.4 Recommending travel itineraries

In tourism, recommending travel itinerary for tourists is a basic but complicated task. When tourists visit a city for several days, there are usually too many attractions to visit in a limited time. Therefore, an intelligent system that can automatically recommend a travel itinerary for tourists becomes an attractive solution. By utilizing the rich information contained in social media data (e.g.

[4] http://www.yelp.com

[5] http://www.wikipedia.org

travelogues, GPS traces, geo-tagged photos etc.), research developed different travel trip planning systems to recommend travel itineraries for users.

Yoon et al. (2010) developed an itinerary recommendation system by utilizing the collective digital trails (e.g. GPS trajectories) provided by experienced travelers. Using 17,745 GPS trajectories recorded by 125 users in Beijing China, their system data mines 119 POIs from detected stay points along these trajectories with a clustering algorithm. A direct graph is then built based on transitions between these POIs. To find top-k efficient travel itineraries given a start/end location and a time budget, the authors developed a heuristic algorithm to compute a candidate trip (sub-graph) by iteratively inserting feasible POI into existing trip. The trip that ends up containing the most classic travel sub-sequences generated by real contributors is selected as the optimal solution.

Xie et al. (2011) advanced the travel itinerary planning problem by developing a composite recommendation system, which can recommend packages of items in sequential form under a given budget of time and money. In their system, POI information such as ratings, monetary cost and location, were extracted from Yelp.com. The system further assume that time spent at a POI is proportional to its size and tourism category. Using an exponential-time orienteering problem solver to find the optimal solution, their system recommends the composite sequence, which reaches a maximum visiting value (rating) of certain POIs subject to a budget constraint. Their system also provides a graphical user interface (GUI) to allow Internet users to dynamically modify the recommended packages (tours) by setting destinations, cost budgets and travel preferences or adding/removing POIs from current solution.

In recent years, geo-tagged photos and the embedded travel information have been used to tackle some simple travel trip planning problems to find and recommend travel itineraries for tourists. De Choudhury et al. (2010) presented a novel methodology to automatically construct travel itineraries from Flickr photos. In their method, individual travel trails are first extracted from Flickr photos, which are mapped to existing POIs and connected as timed paths based on a photo's timestamp, geo-location and textual tags. Based on massive travel trails, they construct an undirected graph structure where each node is a POI with visit time constrains and popularity score and the arc length between nodes represents transit time between two POIs. Then, they treat the mine itineraries problem as an Team Orienteering Problem, and borrow Chekuri et al's (2005) recursive greedy algorithm to discover multi-day itineraries with maximum possible popularity scores in a given time budget. The output of their solution is a text-based daily diary to describe which places to visit, how long to stay and when to transit to the next place. Their user studies demonstrate that this approach can be successfully applied in mining meaningful travel itineraries in 5 tourist cities.

Based on 20 million geo-tagged photos from Panoramio.com and 200,000 travelogues, Lu et al. (2010) developed an online trip planning system, which can not only recommend popular landmarks but also the visiting order and time to spend in each landmark. These landmarks are first mined by using the Mean Shift clustering algorithm, and then mapped to destination names based on gazetteers mined from travelogues. These landmarks are then used to reconstruct travelers' discrete travel paths. Based on these travel paths, the authors developed an Internal Path Discovering (IPD) algorithm to discover classic travel paths within a destination, and a dynamic programming base Travel Route Suggestion (TRS) algorithm to find travel routes among destinations. The IPD works by aggregating multi travel paths within a destination aggregately into several complete travel paths. The TRS is based on a dynamic graph construction. To find travel paths with highest visiting scores for a given time budget, each node in this graph is defined as a possible stay in a destination (e.g. 2 hours stay in place A, 4 hours stay in place A) and is assigned a visiting score (defined by number of visitors) weighted by its possibility. Further, the edge between two nodes is assigned with a score that is defined by the number actual travelers on this edge.

Roy et al. (2011) designed an interactive itinerary planning system. This system can suggest optimized itineraries based on users; feedback on candidate POIs. The POIs are extracted from a Lonely Planet dataset. For a tourist destination, top ranked POIs are first delivered to users for an initial review. Based on user's feedback, candidate POIs will be added to an itinerary based on conditional pair-wise probabilities $\text{Pr}(\text{POI}_i|\text{POI}_j)$, which are derived from Flickr data and presented to users for further review. To construct an optimal itinerary from POIs selected by users, they developed a heuristic greedy itinerary planning algorithm to find the optimal itinerary with best scores for plans that use travelers' time budge efficiently. The expected score of an itinerary is calculated based on users' feedback ("*yes/do_not_care/no*") on POIs and the conditional pair-wise probabilities between POIs.

1.3 Methodology

In this research, I propose an integrated method to build an intelligent tourist trip plan system based on geo-tagged photos from the Internet. This system advances existing systems in several ways. First, for applicable to real-life traveling situation, POIs and related properties, such as attractive score and visiting time etc., are discovered using a density based clustering approach from traveler contributed geo-tagged photos. Besides, the actual trips, driving distances and hours between POIs are used to construct directed traveling graph model. Second, for tackling a multi-day and multi-stay travel planning problems efficiently, an intelligent algorithm is developed to automatically find the approximate optimal travel plan using the discovered POIs and traveling graph model.

The process model of this tourist trip plan system is shown in figure 1.1. In this model, all geo-tagged photos are retrieved automatically from the Internet using information retrieval techniques in the first module. Then, in the next module, POIs are discovered from geo-tagged photos by using a hierarchical clustering algorithm (OPTICS). Based on POI information and clustering results, a traveling graph model is constructed to describe the connectivity among POIs as well as clusters. In the module "Build Tourist Trip Plan System", a meta-heuristic algorithm is designed to find an approximate optimal travel plan given a tourist's travel constraints, which include the start/end locations, the number of traveling days, the time to start/end daily trip and the maximum driving hours between destinations, using the POIs and the traveling graph model. In addition, such traveling graph model can be used to mine travel patterns to gain important travel knowledge in the "Mine Travel Patterns" module. A web-based system of intelligent tourist trip plans is developed for users making their travel plans and viewing travel knowledge interactively and conveniently.

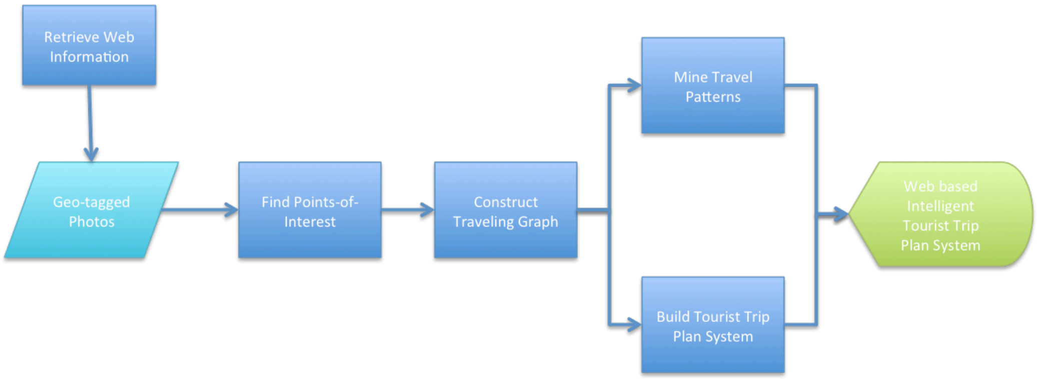

Retrieve Web Information
Geo-tagged Photos
Find Points-of-Interest
Construct Traveling Graph
Mine Travel Patterns
Build Tourist Trip Plan System
Web based Intelligent Tourist Trip Plan System

Figure 1.1: The process model of discovering popular places and movement patterns from geo-tagged photos

1.3.1 Finding POIs from Geo-tagged Photos

Popular places form because most people like to visit them and generate more photos at them than at other places. This leads to an uneven spatial distribution of geo-tagged photos in space (see figure 1.3). Based on this fact, the density-based clustering approach becomes a natural solution to mining these popular places from geo-tagged photos.

In this research, a clustering technique that groups objects into a set of meaningful and useful clusters is applied to find POIs from photo-based POIs. Specifically, I adopt the OPTICS (Ankerst et al. 1999) algorithm to fulfill this task. OPTICS is an extended density-based clustering algorithm that provides a hierarchical clustering structure through an augmented ordering of data points. It is robust to its input parameters and there is no need to identify the number of clusters as input. Since it is a density-based approach, it can filter out the sparsely distributed geo-tagged photos as noise data, and can detect clusters of any arbitrary shape.

The basic idea of OPTICS is to compute the point density around a given point for two input parameters: generating-distance ε and *MinPts*. If there are more than *MinPts* points in the search area with radius equal to ε of point p_i, the *core-distance* of p_i is calculated as the distance to its *MinPts*-th neighbor:

$$core - distance_{\varepsilon,MinPts}(p_i) = \left\|p_i - p_{i,MinPts-th\ nieghbor}\right\|_2 \qquad (1.1)$$

The *reachability-distance* of p_i from any other point p_j in the search area is calculated as the maximum value between the distance of p_i and p_j and the *core-distance* of p_i:

$$reachability - distance(p_i, p_j) = \max\left(core - distance_{\varepsilon,MinPts}(p_j), \left\|p_i - p_j\right\|_2\right) \quad (1.2)$$

Otherwise, both *core-distance* and *reachability-distance* are set to infinity (∞). Then, OPTICS starts from an arbitrary point whose *reachability-distance* is set to ∞ and visits a candidate list of neighbor points based on the rule that the neighbor point with the minimum *reachability-distance* to the current point will be visited next. The value of ε is used to determine the size of the candidate list of neighbor points around p_i.

The results of OPTICS are a set of clusters organized in a tree-like hierarchical structure. All leaf nodes in this tree are the smallest clusters that are used to describe POIs. The parent node of a set of POIs is the region that contains several geographically proximate POIs. Such regions, at different spatial scales (e.g. city/province/country), can be used to provide tourists with abstract travel information (e.g. travel patterns) for getting a good sense of travel places before making travel itinerary recommendations.

To approximately represent any arbitrary area of POI, this research uses the Alpha Shape (Edelsbrunner and Mücke 1992) to describe any convex and concave shape as follows:

$$POI_i = AlphaShape_\alpha(Cluster_i(Photos)) \tag{1.3}$$

where parameter α is the radius of an empty disc. The Alpha Shape approach uses this empty disc to touch point pairs in point space. Given a rational α,the point pairs that were only touched by the empty disc should be mostly located at the margin of the point cluster. Connecting all point pairs will generate an enveloping shape of given points. Different α values will lead to different output alpha shapes. A small α value will generate an empty space, carving out the entire space except for the original points, while a large α value leads the procedure to ignore inner hulls, producing relatively blurry outline sketches of the input points because the large disc size constrains the procedure's ability to traverse and filter multiple candidate points.

1.3.2 Properties of POI

From the clustering results, a POI is basically an area that contains a large enough number of geo-tagged photos taken by tourists. In order to use these POIs to recommend travel itineraries, the POIs need to be assigned popular names and visiting locations, so that tourists knows where and how to visit POIs. To name a POI, this research applies Mean Shift algorithm to the geo-tagged photos within the POI to find the peaks of the geographical distributions of photos. The peak has the highest density value is mapped to and labeled based on the nearest feature found on a preloaded OpenStreetMap POI[6] dataset. This research will use the locations of labeled POIs for the travel itinerary design.

For the purpose of tackling the travel itinerary design problem, additional properties of the POI need to be generated from the POI geo-tagged photos. The first property is the attractive score S_i of a POI. Usually, the goal of travel itinerary design is finding a set of routes that generate a maximum attractiveness score. In this research, an attractive POI score is defined as the number of unique travelers that take photos near the POI. It is computed by using a simple geometry intersection (e.g. point inside polygon) test. This definition is based on an assumption that more attractive place should attract more photographers to visit.

The second POI property for designing travel itineraries is the suggested visiting time T_i. Since tourists have limited time to travel per day, knowing how long to spend in each POI is important for selecting which POI to visit and for

[6] OpenStreetMap POI is a point feature on a map that is not necessary interesting for travel (e.g. post box, car parks etc.), see http://www.openstreetmap.org

making a proper travel itinerary. Fortunately, nearly all geo-tagged photos have timestamps that record when people visited which place. Therefore, it is possible to estimate the suggested visiting time of a POI from the statistics of the timestamps in photos. In this research, for simplification, the suggested visiting time of POI is computed as the average visiting time that photographers spent at the POI. The visiting time of each photographer is extracted as the duration between the first and last photo that a photographer took within the area of POI.

The third POI property for designing the travel itinerary is the time window (e.g. the opening and closing hours of the POI). Different POIs can have different time windows. For example, museums might close earlier than theme parks, and national parks usually have longer operating hours than other types of POIs. Therefore, when making a travel itinerary, the opening and closing time of a POI should also be considered, so that tourists can have enough time to travel efficiently and avoid waiting times or rush hours.

1.3.3 Traveling Graph Model

Since each unique geo-tagged photo can represent the location that a person has visited and has been interested in, individual m's discrete travel route TR_m can be described by connecting the geo-tagged photos in chronologically order:

$$TR_m = Photo_j \xrightarrow{\Delta t_0} Photo_k \xrightarrow{\Delta t_1} \dots \xrightarrow{\Delta t_{n-1}} Photo_n \qquad (1.4)$$

where m is the identifier of user, Δt is the time spent when this user traveled between two photos. As shown in the review above, such digital footprints have been approved useful for mining interesting travel information. This research will take advantage of these digital footprints to build up a traveling graph model for solving the travel itinerary design problem. The basic idea of this traveling graph model is to construct a POI based graph model from individual travel routes based on geo-tagged photos. This is then used to find the optimal POI-based travel routes for a travel itinerary that matches a user's travel requirements.

By using the OPTICS clustering results, each geo-tagged photo can be classified into one of the discovered POIs. Based on that, each individual's photo-based travel routes can be abstracted to a POI-based travel route. Specifically, a travel route can be defined as a sequential structure that connects individual POIs visited in chronological order:

$$TR_m{}' = POI_j \xrightarrow{\Delta t_0} POI_k \xrightarrow{\Delta t_1} \dots \xrightarrow{\Delta t_{n-1}} POI_n \qquad (1.5)$$

Using these reconstructed POI-based travel routes, this research builds a POI-based traveling graph model, which is defined as a directed graph model G:

$$G = (V, E) \qquad (1.6)$$

where node set V are POIs and edge set E are actual travels between two POIs. Each node V_i has an attractiveness score S_i, suggested visiting time T_i, and a time window $[O_i, C_i]$ for describing opening and closing time. Each edge $E_{i,j}$ from node V_i and V_j has two values $\{t_{ij}, w_{ij}\}$, where t_{ij} is the driving time from node V_i to V_j, and w_{ij} is the weight to measure the reoccurrence that travels from node V_i to V_j.

In this research, the traveling time t_{ij} between two POIs is computed based on freely available datasets of the road network from OpenStreetMap. Since computing the distance matrix of POIs based on road networks is extremely expensive in computational terms, this research uses a local version of the open-sourced Open Source Routing Machine (OSRM) system developed by Luxen and his colleagues (2011) to generate shortest driving distance and estimated driving hours between any pair of POIs fast and efficiently. The estimated driving hours are calculated based on the speed limitation information of each road segment. OSRM applies the Contraction Hierarchies technique (Geisberger et al. 2008) and the bi-directional search algorithm (Sint and de Champeaux 1977) to speed up search times for the shortest path between two points on a large-scale road networks dataset.

The weight value w_{ij} f rom node V_i to V_j is defined as the importance and future occurrence of sub-sequence $V_i \rightarrow V_j$ in a travel route. To compute the w_{ij}, I adopt the algorithm of discovering sequential patterns in Association Analysis (Tan et al. 2006) by treating POI-based travel routes as a type of time series data.

Assuming that $DS = \{TR_1, TR_2, \ldots, TR_n\}$ is a dataset that contains n POI-based travel routes extracted from individual geo-tagged photo albums, a travel route then becomes a type of sequence data that may contain many sub-sequences. For example, given $TR_1 = POI_3 \overset{\Delta t_0}{\longrightarrow} POI_1 \overset{\Delta t_1}{\longrightarrow} POI_2$, the sub-sequences of TR_1 can be the set $tr_{1,1} = POI_3 \rightarrow POI_1$, $tr_{2,1} = POI_1 \rightarrow POI_2$, $tr_{3,1} = POI_3 \rightarrow ROI_2$, and $tr_{4,1} = TR_1$. By using the sub-sequence as the basic element, this research computes the support value of each unique sub-sequence to measure its recurrences. The support value of a sub-sequence $tr_{i,j}$ of TR_j is the count of all travel routes that contain $tr_{i,j}$. Given the number of visiting days as time constraint TW, the support value of a sub-sequence can be computed as:

$$support(tr_{i,j}) = \frac{count\{tr_{i,j} \subseteq TR_k \& tr_{i,j}.time \subseteq TW,\ TR_k \in DS\}}{N}, k, j \in [1, N] \quad (1.7)$$

where $tr_{i,j}.time$ denotes the time spent on this sub-sequence. A very low support value of an item means a very low chance you will see it, while a high support value indicates a high chance to see this item. The weight value w_{ij} from node V_i to V_j is a special case of support value of sub-sequences with length 2:

$$w_{ij} = support(V_i \rightarrow V_j) \quad (1.8)$$

1.3.4 Travel Patterns

Using this traveling graph model, useful knowledge, such as travel patterns and behaviors, can be mined at different spatial scales. In this research, classic travel routes can be discovered as top ranked subsequences by their support values from all candidate k-subsequences ($k\epsilon[1, n]$, where n is the maximum length of POI-based travel routes). This is done by iteratively processing candidate k-subsequences where k starts with 2 and increases 1 at the next iteration. In each iteration, all candidate k-subsequences will be generated. The travel routes with support values lower than *min_support* will be removed until there is no candidate.

The travel patterns can be detected and visualized by using travel flow techniques. The major travel flow is defined as a sub-graph G: g where all sub-nodes V_g are connected through sub-edges E_g with a support value greater than a pre-defined support threshold: minimum support value ($min\_support$). Therefore, the task of mining major travel flow is finding all 2-subsequences with a support value larger than $min\_support$. Since OPTICS returns a hierarchical clustering structure, traveling graph models can be generated at different hierarchical levels, which can represent different spatial scales. By applying the same approach, travel patterns can be mined from different spatial scales.

1.3.5 Tourist Trip Design

Tourist trip design is also a classical problem in operations research and has been applied extensively in travel and tourism industry. The basic paradigm of tourist trip design problems is called Orienteering Problem (OP) (Hagen et al. 2005), which is used to solve the problem that, given a set of attractions with visiting scores and a time budget, finding a tour to maximize the collected scores from selected attractions. It is also called the selective travelling salesperson problem (Laporte and Martello 1990), which is a proven NP problem. Many heuristics solutions have been developed to solve this problem in a polynomial time (see a review in Vansteenwegen et al. 2011b). However, the problem described in OP is too simple to handle the complex travel path plan problem in real life. Several extensions of OP are proposed to formulize different and more complex tourist trip design problem.

The team orienteering problem (TOP) is used to formulize multi-day travel path plan by introducing the days of traveling in the problem to OP. In TOP, each member in a team solves an OP for each day without overlapping their selected attractions. The orienteering problem with time window (OPTW) introduces the restrictions (e.g. the opening and closing time) of attractions to mimic the fact that tourists usually need to consider the service hours of visiting places. The team orienteering problem with time windows (TOPTW) is an extension of TOP and OPTW to formulize the travel itinerary plan problem by taking opening and closing hours of attractions into account and allowing for multi-day travel at the same time. Although TOPTW is closer to realistic travel path plan problems, a research gap exists to deal with this problem because of its complexity.

In the TOPTW, every attraction is assigned a visiting score, estimated visiting time and a time window (e.g. opening and closing hours). The target of TOPTW is finding a fixed number of routes, which together contribute the maximum sum of visiting scores. The number of routes is set to the days of visiting. For each day, the trip/route starts and ends at specific times and at the same origination location, and there is no overlapping between visiting time of attractions on the route. Since TOPTW is a difficult combinatorial optimization problem, exact solution approaches, which require long execution times with expensive computation resources to find an optimal solution, are not feasible to apply in real-world applications.

Existing research work develops efficient heuristic approaches to find a suboptimal solution with only a small loss in solution quality. Righini and Salani (2009) proposed a bi-directional and bounded dynamic programming with decremental state space relaxation to tackle the OPTW problem in polynomial time. Montemanniand Gambardella (2009) developed an algorithm based on ant colony system to solve the OPTW that outperforms existed algorithms. Then, Vansteenwegen et al. (2009) developed an Iterated Local Search based meta-heuristic to solve the TOPTW running on mobile devices in real-time. In their following work (Vansteenwegen et al. 2011a), a meta-heuristic based on Greedy Randomized Adaptive Search Procedure (GRASP)(Feo and Resende 1995) is proposed to solve an extension of TOPTW, which allows lunch breaks in a daily trip.

The differences between OP-related research and travel recommendation research based on geo-tagged photos include (1) in OP related research, the attractions are normally artificially designed instances in experiments, or usually known from existed resources (e.g. tourist offices). In contrast, in travel research based on geo-tagged photos, the attractions are discovered from the social media data by utilizing the geographical distributions and densities. (2) in OP-related research, the attractiveness scores of POIs are normally predefined with categorical scores according to the types of attractions (e.g. museum, archaeology, nature etc.), or can be customized categorically by users. In travel research based on geo-tagged photos, the ranking scores of discovered attractions are generated from the photos taken around the attractions. Since research based on geo-tagged photo provides more information for traveling it is more applicable for realistic solutions.

1.3.5.1 Problem Definition

In this research, I define a unique multi-day and multi-stay tourist trip design problem as a multi-stay team orienteering problem with time window. This problem can be treated as an extension of the well-known TOPTW problem by allowing the tourist to start and end a tour at different locations on different travel days. If the destinations in all travel days are the same (e.g. tourist stays in a fixed hotel during travel days), this problem is the same as in TOPTW. To allow the daily tour to end at a different location than the start location (e.g. a tourist can stay in different hotels during travel days), I assume that a tourist can always find a hotel for accommodation within or near any POI after completing the tour.

In multi-destination TOPTW, a set of N POIs, each is assigned with an attractiveness score S_i, suggested visiting time T_i, and a time window $[O_i, C_i]$ for describing opening and closing time, is given. From POI$_i$ to POI$_j$, the driving time t_{ij}, the maximum driving time t_{max}, and the trip reoccurrence weight w_{ij} are also given. Every POI can be visited at most once. The goal of multi-destination TOPTW is, given a start location, end location, a daily time budget T_{max}, and the number of traveling days M, to find a travel itinerary that maximizes the total attractive score by visiting selected locations under the given daily time budget in a given number of travel days. This multi-destination TOPTW problem can be formulated as an integer program in mathematics as follows:

$$Max \sum_{d=1}^{M} \sum_{i=2}^{N-1} S_i y_{id} \tag{1.9}$$

subject to:

$$\sum_{j=2}^{N} x_{1j1} = \sum_{i=1}^{N-1} x_{iNM} = 1 \tag{1.10}$$

$$\sum_{i=1}^{N-1} x_{ikd} = \sum_{j=2}^{N} x_{kjd} = y_{kd} \quad \forall k = 2, \dots, N-1; d = 1, \dots, M \tag{1.11}$$

$$s_{id} + T_i + t_{ij} - s_{jd} \leq U\left(1 - x_{ijd}\right) \ \forall i, j = 1, \dots, N; d = 1, \dots, M \tag{1.12}$$

$$\sum_{d=1}^{M} y_{kd} \leq 1 \ \forall k = 2, \dots, N-1 \tag{1.13}$$

$$\sum_{i=1}^{N-1}\left(T_i y_{id} + \sum_{j=2}^{N} t_{ij} x_{ijd}\right) \leq T_{max} \quad \forall d = 1, \dots, M \tag{1.14}$$

$$t_{ij} x_{ijd} \leq t_{\max} \qquad \forall i = 1, \dots, N; d = 1, \dots, M \tag{1.15}$$

$$O_i \leq s_{id} \quad \forall i = 1, \dots, N; d = 1, \dots, M \tag{1.16}$$

$$S_{id} \leq C_i \quad \forall i = 1, \dots, N; d = 1, \dots, M \tag{1.17}$$

$$x_{ijd}, y_{id} \in \{0,1\}; \quad \forall i, j = 1, \dots, N; d = 1, \dots, M \tag{1.18}$$

where y_{id} means that POI$_j$ is visited in d-th day; s_{id} is the start time of visiting POI$_j$ in d-th day; U is a large constant which is set to the largest positive 4-bytes integer; x_{ijd}=1 represents that, on the d-th day, there is a visit of POI$_j$ after POI$_i$, otherwise x_{ijd}=0. The objective function (1.9) is to maximize the total attractive score in M days. Constraint (1.10) makes sure the entire tourist journey starts from 1st POI 1 and ends at Nth POI. The start can be the same as the end. Constraint (1.11) ensures the connectivity of the whole tour. Constraint (1.12) ensures the starting visiting time of a selected POI is feasible. Constraint (1.13) ensures that each POI is visited at most one time. Constraint (1.14) ensures that the total traveling time in each day is within a given time budget. Constraint (1.15) ensures that the driving time between two POIs does not exceed the maximum allowed driving time t_{max}. Constraints (1.16) and (1.17) ensure that visiting a

POI occurs between its opening and closing time. Constraint (1.18) denotes this is an integer program.

1.3.5.2 Heuristic Solution

In this research, the proposed multi-stay TOPTW that extends from TOPTW is a highly constrained problem, so that the optimal solution can not be solved within polynomial time by using exact solutions, such as integer programming. Existing research on classic TOPTW proved that developing efficient heuristic approaches can be used to find a suboptimal solution quickly with only a small loss in solution quality. Based on the existing meta-heuristic solutions of TOPTW, this research proposes a modified Iterated Local Search (ILS) heuristic algorithm to efficiently find approximate optimal solution of this problem by using additional travel information, such as attractive score of POI, reoccurrence weights of trip etc., mined from geo-tagged photos.

The proposed ILS-based heuristic executes a limited number of local searches iteratively to generate a set of local solutions sequentially to find the best solution. Each ILS includes two major steps: Construct step and Shake step. The construct step adds new feasible visits to a tour. At each time, the feasible visit, which brings the highest visiting benefit (see definition in formula 1.19) when adding it to a current tour, will be selected and inserted into a current tour until there is no available time for any extra visit. The shake step removes one or more visits in a current tour, and then in a next iteration, the insertion step will seek and add different feasible visits to generate a different tour. The rule of removing visits from a current rule is defined in the shake step, which is to ensure that every visit is removed at least once. By doing so, the heuristic can escape local optima and better explore the entire solution space to approach a possible optimal solution. The local search will end when the current best solution found so far is not improved in a predefined number of iterations.

1.3.5.2.1 Construct step

This heuristic algorithm starts with an initialized tour where the given start location and end location are placed at the two ends of the tour. M-1 virtual POIs are placed equally in the tour to represent the places of accommodation. The virtual POI is a dummy POI that has no location information but has time information to indicate when to start a day trip (e.g. 8am). Then, each tour segment that is divided by virtual POIs represents a one-day trip (see figure 1.2).

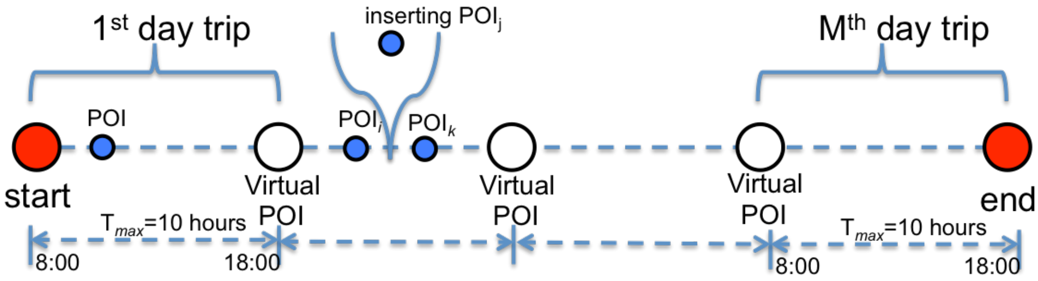


Figure 1.2: An illustration of the construct step in the proposed heuristic algorithm for multi-day and multi-stay travel trip plan

To decide which POI is the best candidate to insert into a current tour, for each POI that has not been selected for visiting and can be reached within the predefined maximum driving time t_{max}, a inserting benefit score is calculated at every feasible position in every tour segment. The inserting benefit of POI$_j$ between POI$_i$ and POI$_k$ is defined as following:

$$Benefit_{j,ik} = \frac{s_j{}^2(w_{ij}+w_{jk})}{ExtraTime_{j,ik}} \quad (1.19)$$

where $w_{ij} + w_{jk}$ is the weighted of attractiveness score of POI$_j$ based on the actual travel data that extracted from geo-tagged photos. The more reoccurring travel between POI$_i$, POI$_j$ and POI$_k$, the more weight is assigned to the attractiveness score. $ExtraTime_j$ represents the extra time to be consumed when inserting POI$_j$ between POI$_i$ and POI$_k$. It is defined as follows:

$$ExtraTime_{j,ik} = t_{ij} + Wait_j + T_j + t_{jk} - t_{ij} \quad (1.20)$$

where $Wait_j$ means possible waiting time at POI$_j$ before its opening time because of early arrival, and it is defined as:

$$Wait_j = \max(0, O_j - arrive_j) \quad (1.21)$$

where $arrive_j$ means the arrival time of POI$_j$. Greater benefits of a tour are associated with less time needed for visiting more attractive candidate POIs.

Calculating benefits for candidate POIs inserting at all possible positions on a tour would be computationally expensive, especially when the size of candidates and number of given travel days are large. To restrict the size of candidate inserting after one POI, the k-nearest candidate POIs that are within the distance of t_{max} driving time are queried by using a kd-tree index.

To speed up the evaluation of possible insertion at different positions in a tour, for each selected POI in every tour segment, the maximum extra time that is allowed for visiting a candidate POI in the travel dayis recorded to evaluate candidates fast. For selected POI i, the maximum extra time is defined following:

$$MaxExtra_i = \min(C_i - s_i - T_i, Wait_{i+1} + MaxExtra_{i+1}) \quad (1.22)$$

where $C_i - s_i - T_i$ represents the limited time left for visiting other candidates after ensuring a successful visit of the POI itself, which starts at s_i and needs T_i time. When the sum of $Wait$ and $MaxExtra$ time of the next POI in the same tour segment is less than its own maximum extra time, it will use the sum value of next POI to satisfy the lower boundary case first. Therefore, when inserting a candidate POI$_j$ between POI$_i$ and POI$_k$, the extra time should be less than the maximum extra time allowed by the next POI$_j$ plus a possible waiting time at POI$_j$:

$$ExtraTime_{j,ik} \leq Wait_k + MaxExtra_k \tag{1.23}$$

The candidate POI at the candidate position with the highest inserting benefit will be inserted into the current tour. After inserting the new POI in a tour segment, in this tour segment, all selected POIs after this new POI will update the arrival time $arrive_i$, starting visiting time s_i, possible waiting time $Wait_i$ and maximum extra time allowed $MaxExtra_i$, while all selected POIs before this new POI will just update the maximum extra time allowed $MaxExtra_i$ since inserting a new POI will reduce the maximum extra time allowed in the tour segment. When there is no candidate POI that can be inserted into the current tour within the maximum extra time allowed by each tour segment, a local search is completed and a temporal solution is generated in this iteration. The sum of attractive scores of the current tour is recorded for determining the best solution in all iterations.

1.3.5.2.2 Shake step

Before entering the next iteration, a shake step is applied to remove a set of selected POIs from each tour segment. This research follows the shake heuristic developed by Vansteenwegen et al. (2009) for TOPTW. It has been proved to be a good technique to explore the entire solution space and correct earlier mistaken decisions. There are two parameters in the shake heuristic to determine how many POIs will be removed from where in each tour segment: one is the start location $Start_{tour}$ and the other is the number of consecutive POIs to be removed $Number_{tour}$. The two parameters are set to 0 and 1 initially in the first iteration. After the construction step, the first POI in every tour segment will be removed according to these two parameters. In the next iteration, the number to be removed in each tour segment will increase by 1:

$$Number_{tour} = Number_{tour} + 1 \quad (1.24)$$

while the start location will increase $Number_{tour}$:

$$Start_{tour} = Start_{tour} + Number_{tour} \quad (1.25)$$

The $Number_{tour}$ POIs that start at $Start_{tour}$ will be removed in every tour segment in the following iteration. When $Start_{tour}$ becomes larger than the length of the smallest tour segment, it will be reduced by this length:

$$Start_{tour} = Start_{tour} - \min(length_{tour}) \quad (1.26)$$

When $Number_{tour}$ reaches the maximum number of POIs to remove $N/(3 * M)$, it will reset to 1. As indicated by authors in (Vansteenwegen et al. 2009), this heuristic could ensures that every POI inserted on the tour is removed at least once.

When iterations are completed after the shake step, the heuristic will enter the next iteration so that the construct step can insert new POIs based on the previous trimmed tour to generate a different solution. By doing so, the construct and shake steps work together in the proposed ILS based heuristic, continuously searching for the best solution based on the current solution until the best solution does not get updated in a predefined number of rounds. The details of this proposed heuristic algorithm is described in algorithm4.

1.4 Experiments

1.4.1 Data

I choose Australia as a study area for this research since tourism is a major economic industry in this country that attracts hundreds of thousands of tourists every year. I apply the method presented above to build a travel plan system based on geo-tagged photos to discover attractive places and travel patterns and recommend high quality multi-day and multi-stay travel plans for tourists. To do so, I first crawl all geo-tagged photos that have been geo-tagged in Australia from the website Panoramio.com. A total of 118,736 geo-tagged photos were retrieved from 4,920 registered Internet users of Panoramio.com. On average, each user contributes 24 geo-tagged photos. All photos were taken between 2005 and 2011. The geographical distribution of these geo-tagged photos is shown in figure 1.3.

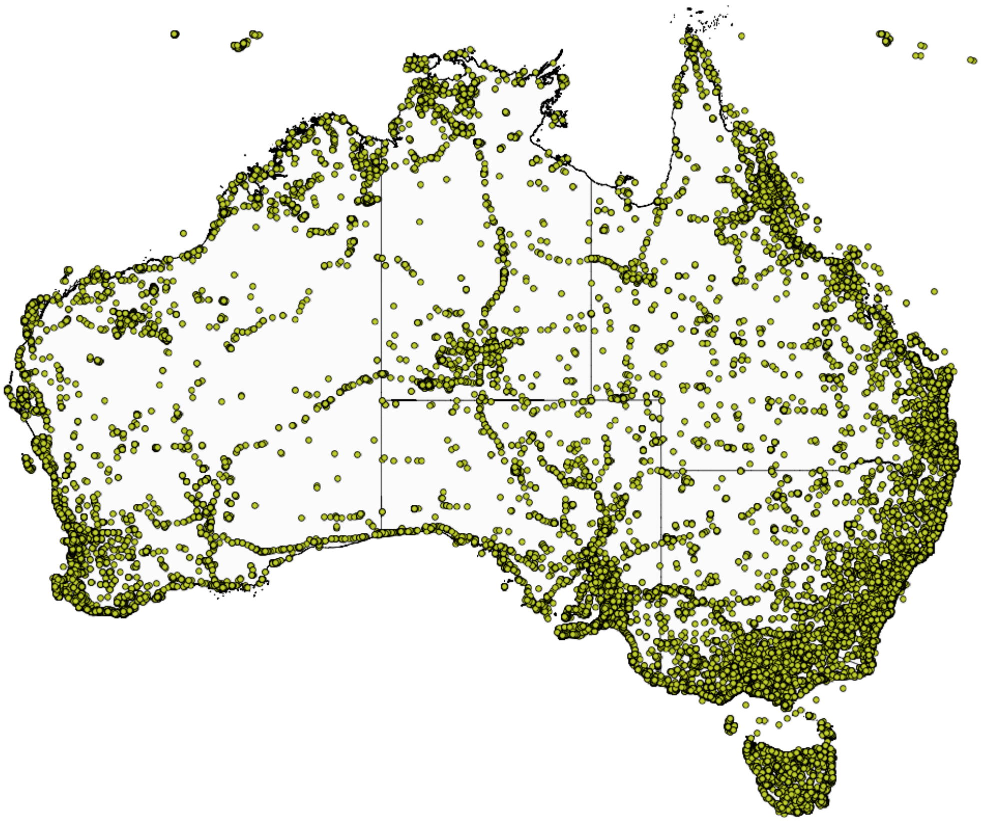


Figure 1.3: Geographical distribution of 118,736 geo-tagged photos that used in this case study of Australia

1.4.2 POIs based on geo-tagged photos

In this experiment, the clustering objects are a large number of geo-tagged photos. Therefore, a large search radius ε will result in expensive runtime cost of OPTICS. It is essential to determine an optimal value for ε in a large data space D that has N points. In this test case, following what Ankerstand colleagues suggested in (1999), I use the expected k-nearest neighbors method, which assumes that all points are randomly distributed in space, to estimate the optimal value of ε that can guarantee a certain number of points can be searched by any core object. Based on this theory, the radius ε of subspace S, which contains exactly k ($k = MinPts$) points in an N points dataset, is calculated by using the following formula:

$$\varepsilon = \sqrt[d=2]{\frac{V_N \times k \times \Gamma(1+d/2)}{N \times \sqrt{\pi^d}}} \tag{1.27}$$

where $\Gamma \cdot$ is the Gamma-function, and V_s is the volume of subspace S. In this case, the volume is the area of the maximum enclosing rectangular of points in S. Experimentally, the parameters *MinPts* is set to 50 and ε is computed equals to 0.431 for running OPTICS.

The running time of the OPTICS algorithm is $O(n * \varepsilon - neighborhoodquery)$, which depends heavily on the running time of the ε-neighborhood query. Therefore, to accelerate finding neighborhoods in the ε search area, I built a kd-tree based on geo-tagged photo data and use the k-nearest neighbor (*KNN*) search algorithm (Mount and Arya 1997) to query the ε neighborhoods in OPTICS. By doing so, the average run time of this OPTICS algorithm can be reduced to $O(n * logn)$.

Unlike traditional agglomerative hierarchical clustering algorithms that produce a tree-like hierarchical structure in the form of dendrograms, OPTICS generates a “reachability plot” Ankerst et al. (1999) where hierarchical clusters are not explicit and need to be extracted from the “dents” separated by spike bars in the plot (see figure 1.4). If drawing a horizontal line (see red horizontal line in figure 1.4) crosses valleys in the plot, each valley underneath the horizontal line can be interpreted as a cluster. The points above this line will be ignored as noise. Moving down the red line, more clusters will emerge as the line crosses more spikes and generates more small valleys. These small valleys and their large valley containers form a natural hierarchical relationship.

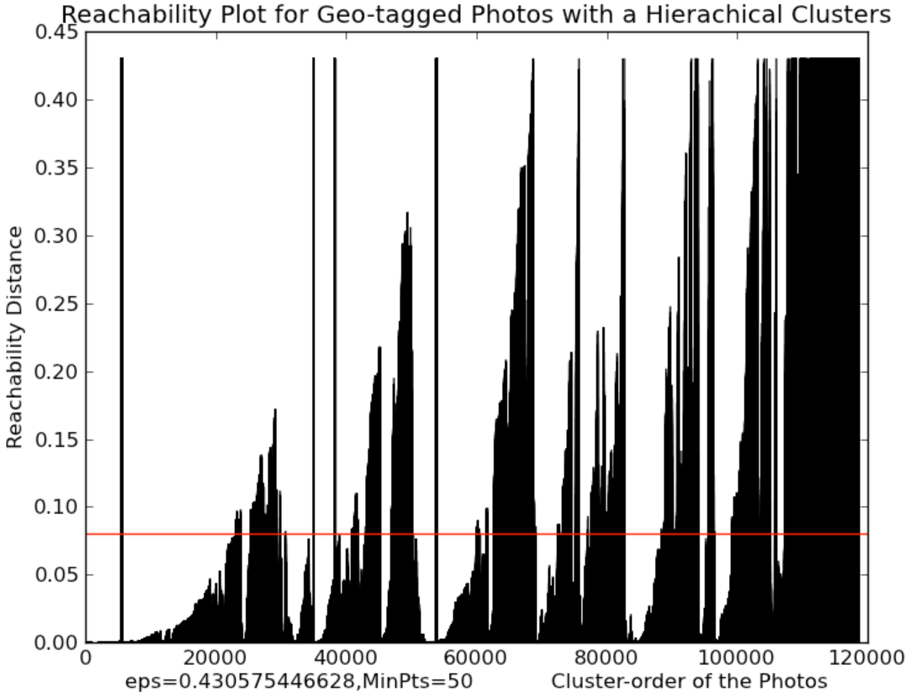


Figure 1.4 Reachability plot of OPTICS clustering results on geo-tagged photos. The red line is a demonstration for finding DBSCAN clusters at different hierarchies.

To automatically extract the hierarchical clustering structure from the OPTICS reachability plot, I use the automatic techniques proposed by Ankerst et al. (1999) to convert the reachability plot to a dendrogramand discover POIs from all leaf nodes in the tree structure (see algorithm 2). Figure 1.5 is a dendrogram that is generated based on the OPTICS reachability plot in figure 1.4. In this tree structure, every leaf node is the smallest POI. A group of POIs that have the same ancestor at a certain hierarchical level can be treated as regions-of-interest (ROIs) at a large spatial scale (e.g. city, province or country).

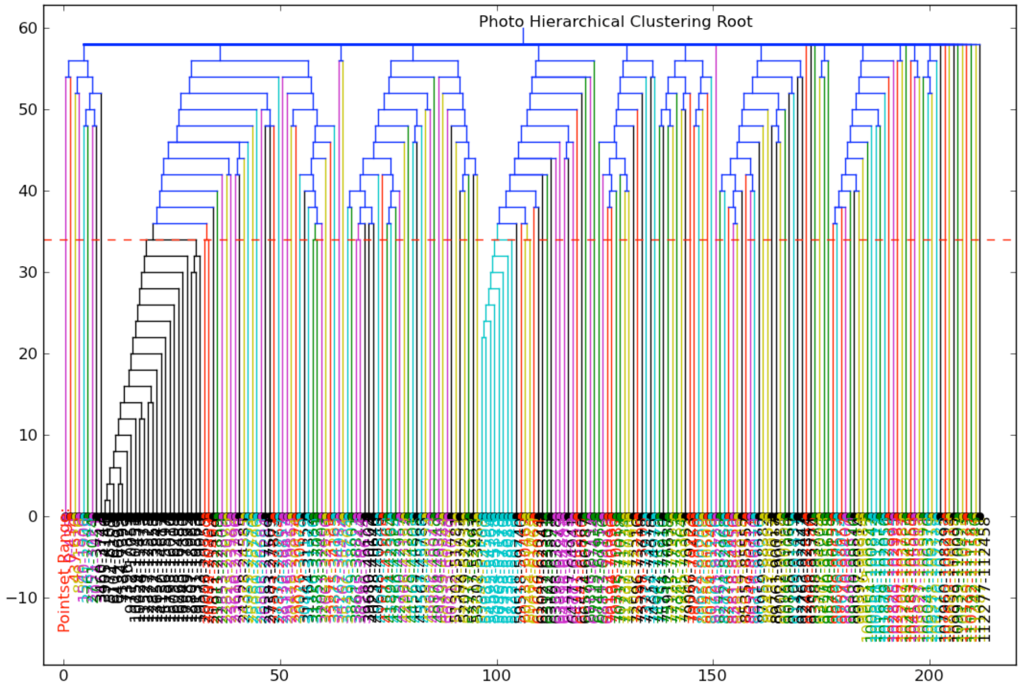
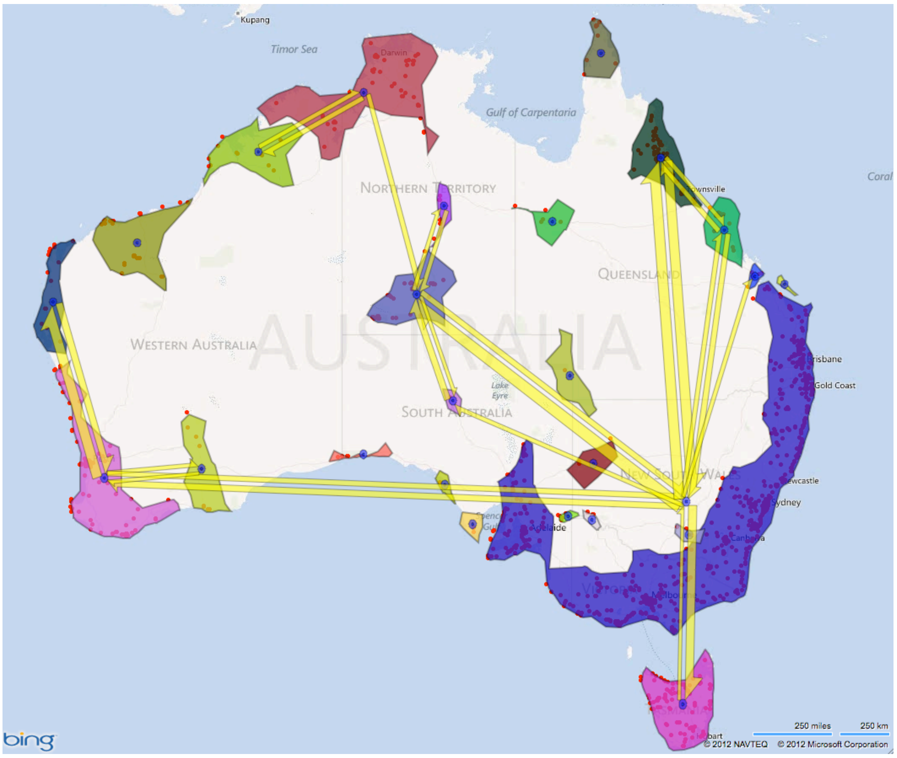
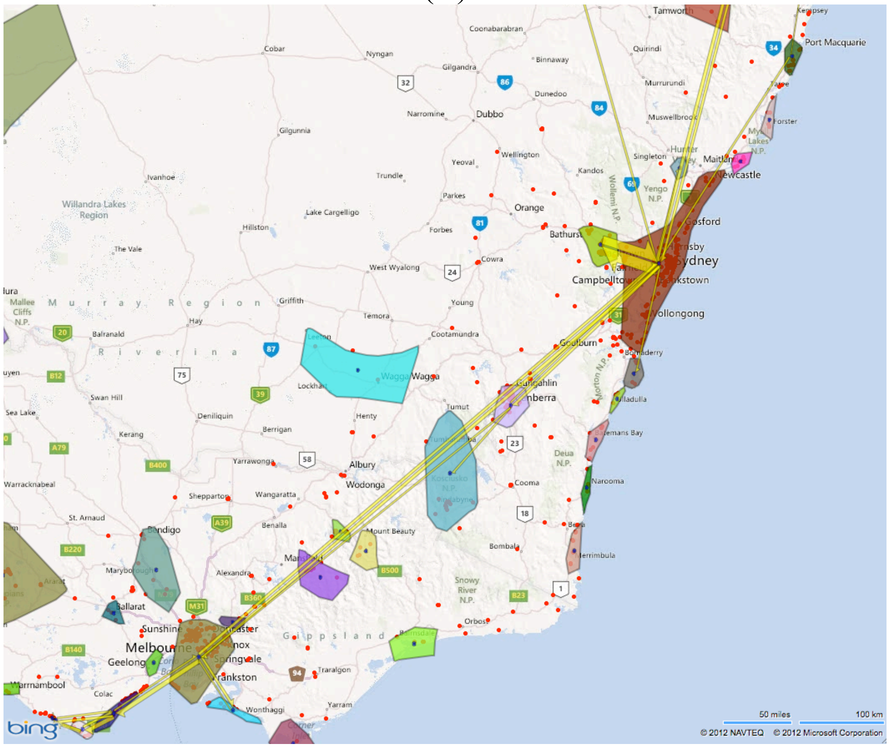


Figure 1.5 The dendrogram that was generated based on the OPTICS reachability plot with real geo-tagged photos.

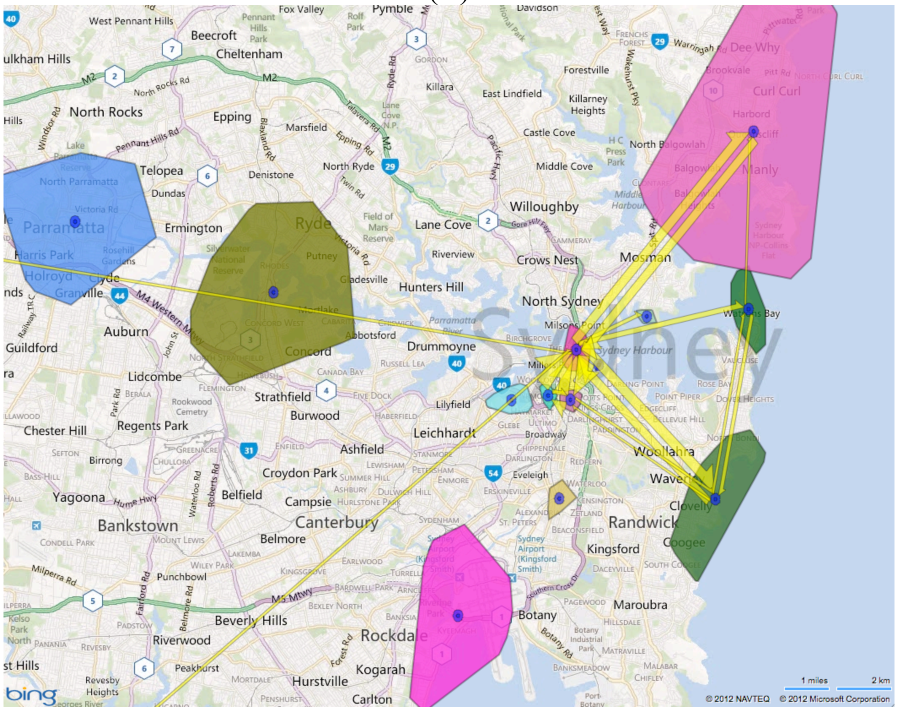
In this empirical experiment, a prototype based on web mapping is developed to visualize the experimental results (e.g. clustering, travel patterns, and travel itinerary). It allows users to explore results at different levels of detail in a scalable online map system. In this system, each POI is visualized using Alpha Shape techniques. Experimentally, I set the parameter α of Alpha Shape equals to the search radius ε of the OPTICS algorithm. The geovisualization of clustering results can be seen in figure 1.6. Figure 1.6(A) shows the POI discovered at the highest hierarchical level 1 (country scale). Figure 1.6(B) shows the POI set, which is shown in a map zoomed to the province scale, detected at hierarchical level 9. Figure 1.6(C) shows the POI set discovered at lowest hierarchical level 30. It is displayed in a map zoomed to the city scale. The detected POIs are displayed as the red dots in figure 1.6(A).



(A)



(B)



(C)

Figure 1.6 (A) (B)(C) POIs, ROIs and travel patterns (yellow arrow flows) that are mined from geo-tagged photos and displayed at different spatial scales in Australia: (A) country (B) province (c) city.

1.4.3 Travel patterns based on geo-tagged photos

By leveraging the geo-tagged photos and the clustering results, a traveling graph model is constructed for discovering the travel patterns as useful for tourism knowledge. The algorithm of discovering sequential patterns in Association Analysis is used to calculate a probability of reoccurrence of traveling between two ROIs. Directional travel routes, with a reoccurrence support value that is larger than a predefined threshold (equals to 0.13 in this experiment) are treated as important routes. By using Tobler's (1987) flow mapping techniques, the prioritized travel patterns discovered by the proposed system can be visually in aggregate form (see figure 1.6).

In the figure 1.6 (A)(B)(C), the arrows in yellow represent the travel flow generated from an individual's digital footprints using their geo-tagged photos. Thicker flow means higher travel volume and reoccurrence in overall travel, while thinner flow means less travel volume. The travel flows that are lower than a predefined threshold are not displayed in order to highlight the important information on the map. Again, since the prototype system allows users to explore different levels of detail in the experimental results of a scalable online map system. Thus tourists can quickly gain the tourism knowledge they need at different geographical scales by interactively examining travel patterns. For example, figure 1.6(A) presents travel patterns at a country scale, (B) shows the travel patterns at a city level (by zooming to a specific area), and (C) displays travel patterns at city level.

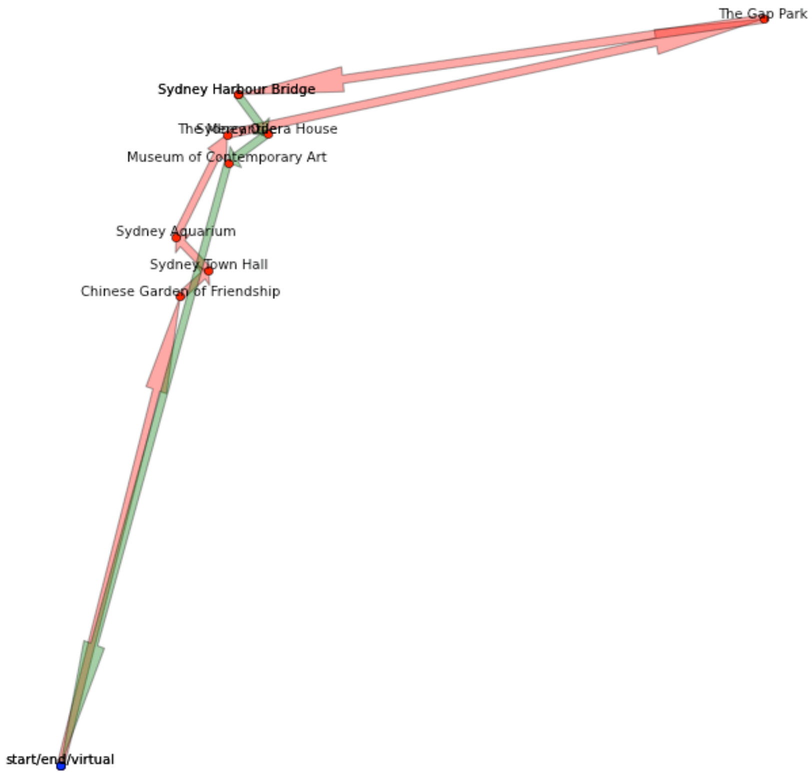
1.4.4 Travel Path Plan

In this empirical experiment, the proposed travel path plan algorithm is tested based on 2,135 POIs and the traveling graph model is discovered from geo-tagged photos. Each POI has an attractiveness score and a suggested visiting time, which are also extracted from the geo-tagged photos. The time window of a POI is setup according to the category of POIs: all parks are simply set to have opening and closing time from 00:00 to 24:00; for all other types of attractions, this test case simply setup their opening/closing hours to 8:00 am/5:00 pm. This system also allows the user to change the service time for any specific POI in configuration.

I design two test cases for testing the travel path plan: the first test case assumes a tourist flies to Sydney International Airport, Sydney, New South Wales, Australia, and has two days available for a trip. When she finishes the travel, she will come back to the Sydney International Airport to fly back home. The second test case is similar to first one, but has 4 days for travel. In these two test cases, tourists are assumed to only plan for a road trip (by car, no plane), their daily trip is restricted to start no earlier than 8:00 am and end no later than 6:00 pm (10 hours quota per day), and the configurable maximum driving time between two POIs is set to default 3 hours.

The results of the test cases are shown in figure 1.7 and 1.8. The plots on the top are abstract illustration of recommended travel paths: different colors represent travel routes on different travel days. The plots on the bottom are corresponding map views of recommended travel paths where the detailed driving routes and the visited POI are displayed in Bing Maps. The plots below are textual travel itineraries to describe the details about where to visit and how long to stay during traveling. These good quality travel path plan results demonstrate the feasibility of the proposed heuristic algorithm for travel path planning.

Both experiments can return solutions within 1 minute, which needs further optimization to get better performance. Since the program is written in Python and similar C++ based TOPTW program developed by Vansteenwegen et al. (2009) can find solutions with hundreds POIs within several seconds, there is a potential that refactorize current code using C++ could speed up this program to second level.


The Gap Park
Sydney Harbour Bridge
Sydney Opera House
Museum of Contemporary Art
Sydney Aquarium
Sydney Town Hall
Chinese Garden of Friendship
start/end/virtual

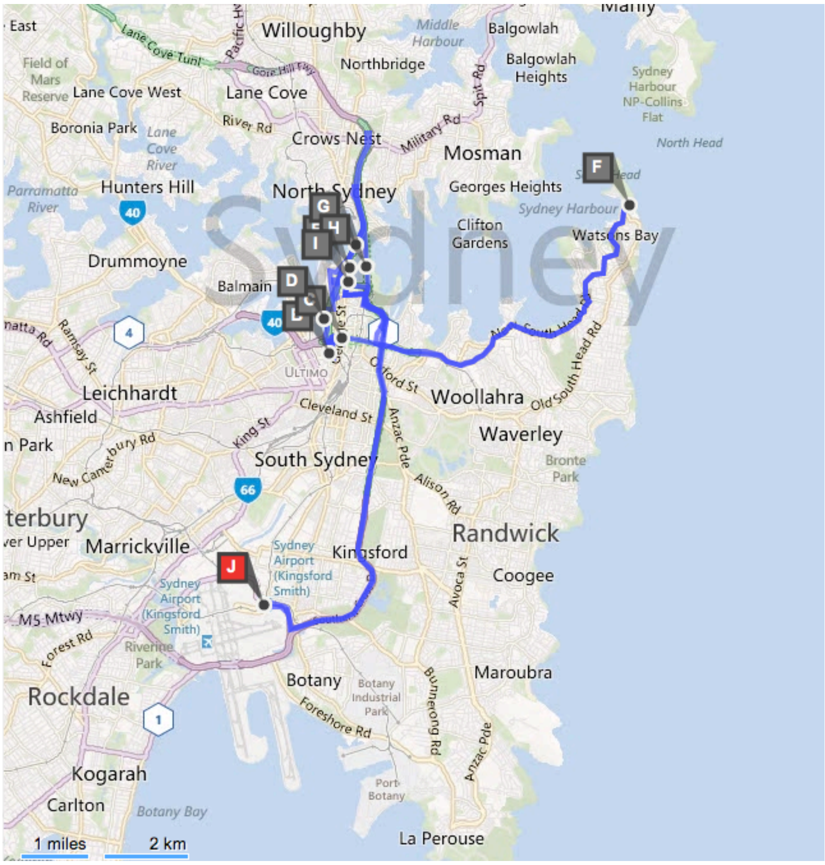

Willoughby
Middle Harbour
Balgowlah
Balgowlah Heights
Manly
Northbridge
Lane Cove West
Lane Cove
Field of Mars Reserve
Boronia Park
Lane Cove River
Crows Nest
Military Rd
Spit Rd
Sydney Harbour NP-Collins Flat
North Head
Mosman
Georges Heights
Hunters Hill
Parramatta River
North Sydney
Clifton Gardens
Sydney Harbour
Watsons Bay
Drummoyne
Balmain
Sydney
Ultimo
Leichhardt
Ashfield
Woollahra
Waverley
Cleveland St
King St
Oxford St
Anzac Pde
South Sydney
Bronte Park
Alison Rd
Randwick
Marrickville
Kingsford
Coogee
Sydney Airport (Kingsford Smith)
M5 Mtwy
Forest Rd
Riverine Park
Rockdale
Botany
Botany Industrial Park
Foreshore Rd
Maroubra
Kogarah
Carlton
Botany Bay
Port Botany
La Perouse
1 miles
2 km
A
C
D
F
G
H
I
J
L

Figure 1.7 A 2-day tourist trip itinerary, which starts and ends at Sydney International Airport.

The detail of the 2-day tourist trip itinerary is shown below:

- Day 1 (pink route):

  start from Sydney International Airport at 8am;

  drive about 0.12 hours to Chinese Garden of Friendship at 8:20, spend 1 hour there;

  drive0.01 hours to Sydney Town Hall at 9:30, spend about 2.4 hours there;

  drive 0.01 hours to Sydney Aquarium at 12:00, spend about 1.5 hours there;

  drive 0.03 hours to the Mercantile at 13:40, spend 3.2 hours there;

  drive 0.15 hours to the Gap Park at 16:50, spend 1 hour there;

  drive 0.14 hours to Sydney Harbor Bridge at 18:00, find a hotel nearby to stay.

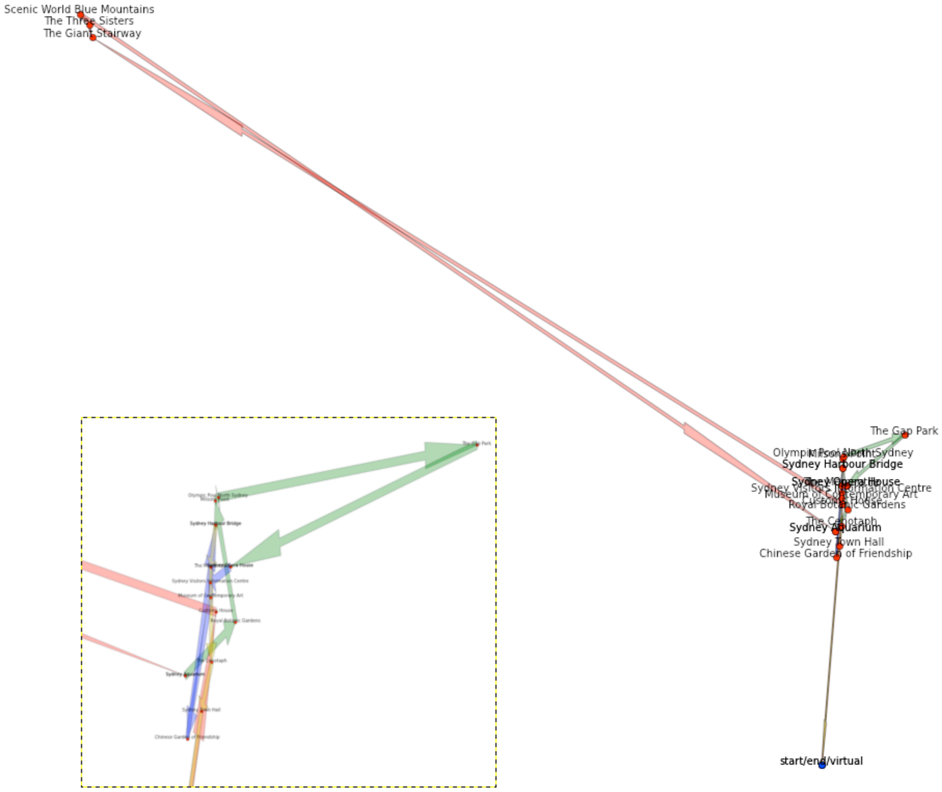
- Day 2 (green route):

  start from near Sydney Harbor Bridge at 8am, spend about 3.9 hours there;

  drive 0.01 hours to Sydney Opera House at 11:50, spend about 3.9 hours there;

  drive 0.01 hours to Museum of Contemporary Art at 15:30 and spend about 1.9 hours there;

  drive 0.15 hours to Sydney International Airport at 18:00.


Scenic World Blue Mountains
The Three Sisters
The Giant Stairway
The Gap Park
Olympic Pool
North Sydney
Sydney Harbour Bridge
Sydney Opera House
Sydney Visitor Information Centre
Museum of Contemporary Art
Royal Botanic Gardens
The Cenotaph
Sydney Aquarium
Sydney Town Hall
Chinese Garden of Friendship
start/end/virtual

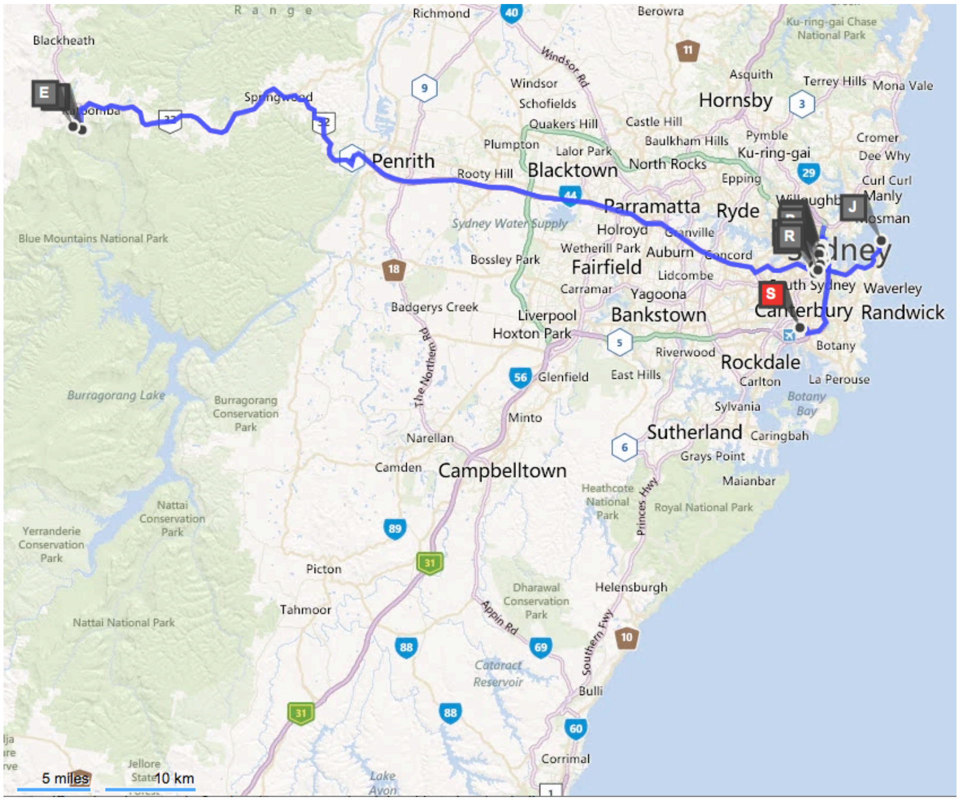

E
J
R
S
Blackheath
Katoomba
Springwood
Penrith
Richmond
Windsor
Schofields
Quakers Hill
Castle Hill
Hornsby
Berowra
Asquith
Terrey Hills
Mona Vale
Ku-ring-gai Chase National Park
Plumpton
Lalor Park
Baulkham Hills
Pymble
Ku-ring-gai
Cromer
Dee Why
Rooty Hill
Blacktown
North Rocks
Epping
Curl Curl
Manly
Mosman
Parramatta
Ryde
Sydney Water Supply
Holroyd
Granville
Wetherill Park
Auburn
Concord
Sydney
Bossley Park
Fairfield
Lidcombe
Carramar
Yagoona
Waverley
Randwick
Canterbury
Badgerys Creek
Liverpool
Bankstown
Hoxton Park
Riverwood
Botany
Rockdale
Carlton
La Perouse
Glenfield
East Hills
Sylvania
Botany Bay
Sutherland
Caringbah
Grays Point
Minto
Narellan
Camden
Campbelltown
Maianbar
Heathcote National Park
Royal National Park
Princes Hwy
Blue Mountains National Park
Burragorang Lake
Burragorang Conservation Park
Nattai Conservation Park
Yerranderie Conservation Park
Nattai National Park
Picton
Tahmoor
Dharawal Conservation Park
Helensburgh
Appin Rd
Southern Fwy
Cataract Reservoir
Bulli
Corrimal
Lake Avon
The Northern Rd
Windsor Rd
Jellore State Forest
R a n g e
5 miles
10 km

Figure 1.8: A 4-day tourist trip itinerary, which starts and ends at Sydney International Airport.

The detail of the 4-day tourist trip itinerary is shown below:

- Day 1 (pink route):

  start from Sydney International Airport at 8am;

  drive 0.15 hours to Customs House at 8:15, spend about 4.5 hours there;

  drive 1.8 hours to The Giant Stairway at 14:45, spend about 1hour there;

  drive 0.01 hours to The Three Sisters at 15:40, spend about 1.5 hours there;

  drive 0.02 hours to Scenic World Blue Mountains at 16:50, spend about 1 hour there;

  drive 1.8 hours to Sydney Aquarium, and find a hotel nearby to stay

- Day 2 (green route):

  start from Sydney Aquarium at 8am, spend about 1.7 hours there;

  drive0.03 hours to Royal Botanic Gardens at 9:50, spend about 1.6 hours there;

  drive 0.03 hours to Milsons Point at 11:20, spend about 2.5 hours there;

  drive 0.01 hours to Olympic Pool North Sydney at 13:50, spend about 2.5 hours there;

  drive 0.15 hours to The Gap Park at 16:35, spend about 1 hour there;

  drive 0.14 hours to Sydney Opera House, and find a hotel nearby to stay

- Day 3 (blue route):

  start from Sydney Opera House at 8am, spend 4 hours there;

  drive 0.01 hours to Sydney Visitors Information Centre at 12:00, spend about 4 hours there;

drive 0.01 hours to Museum of Contemporary Art at 16:20, spend about 1 hours there;

drive 0.03 hours to Chinese Garden of Friendship at 17:20, spend about 0.5 hour there;

drive 0.05 hours to Sydney Harbour Bridge, and find a hotel nearby to stay

- Day 4 (light yellow route):

  start from Sydney Harbour Bridge at 8am, spend about 3.5 hours there; drive 0.01 hours to the Mercantile at 11:30, spend about 3.2 hours there; drive 0.02 hours to the Cenotaph at 14:50, spend about 0.8 hours there; drive 0.01 hours to the Sydney Town Hall at 15:30, spend about 2.3 hours there;

  drive 0.13 hours to Sydney International Airport at 18:00.

1.5 Conclusion

In this essay, I presented a methodology for building an intelligent tourist trip plan system based on online geo-tagged photos. First, I applied information retrieval techniques to collect over one hundred thousand publicly available geo-tagged photos and related metadata from Panoramio.com. Second, using a density-based clustering algorithm (OPTICS), I discovered the attraction regions and POIs with useful travel information (e.g. attractiveness score, suggested visiting time etc.) from the geo-tagged photos. Third, I constructed a traveling graph model to represent the connectivity among POIs and attractive regions. Then, travel patterns were mined by using the traveling graph model and the attractive regions over a wide range of spatial scales. Fourth, I developed an efficient Iterated Local Search based heuristic algorithm to find an approximate optimal solution to the multi-day and multi-stay tourist trip plan problem. I demonstrated the efficiency and utility of this approach by representing travel patterns and finding travel itineraries for tourists using the knowledge discovered from geo-tagged photos in a case study application of Australia.

This research has potential broader impacts for tourism, location-based services, behavioral geography and other fields. For tourism research and practice, this work provides a new solution for discovering useful travel knowledge and recommending travel itineraries based on the online geo-tagged photo collections, which contain rich social media data. This work also tackles the multi-day and multi-stay tourist trip plan problem by developing an efficient heuristic algorithm. It can be used to make customized travel plans for personal guide services. In location-based services, this work can be used to provide valuable tourist services (e.g. real-time tourist trip plan) on GPS-enabled mobile devices. For behavioral geography, this work leverages a new type of behavioral data source for studying human movement behaviors. The travel knowledge discovered from online geo-tagged photos is useful for examining and testing behavioral theory.

Future work includes extending this work to generate user-friendly tourist maps for Internet users generating customized and applicable travel itineraries. To achieve this target, some systems that automatic generate destination maps (Kopf et al. 2010) or tourist maps (Grabler et al. 2010) can be integrated into this proposed system to deliver users with customized thematic travel maps, which can display selected relevant travel routes and layout important 3D POIs nearby, for better spatial cognition. Allow user to add personal events, such as lunch break or naptime, to current travel itinerary is another important future work for applicability. This can be implemented by using virtual POI in proposed algorithm, but requires the algorithm can update or recomputed rest of travel itinerary on the fly. There is also a potential to apply the proposed method further to other forms of spatiotemporal social media data, such as geographically explicit GPS trajectories or location implicit social network data. Integrating different data sources, this work could discover richer travel information and knowledge that can improve the quality of recommended travel itineraries.

At the same time, the proposed methodology exhibits several limitations and remaining challenges for future work. One big challenge is the scalability of the proposed approach. In this work, I tested the proposed methodology by using a small subset of retrieved geo-tagged photos: travel knowledge is discovered from 118,736 geo-tagged photos and the heuristic solution for making a tourist trip plan is tested based on 2,136 discovered POIs. The designed case study principally supports the claim that the proposed scheme can be used to build an efficient tourist trip plan system from digital photographs. However, it is not feasible to directly apply the methodology in this essay on a global scale since the overall data contain about 36 millions records which are much larger than the case study data in this essay. Therefore, to overcome this issue, more efficient data structures and algorithms are still needed.

Another limitation is that the discovered popular places and tourist patterns are not representative of the whole population but are based on Internet users with intent to share their geo-tagged photos. Further, even these specialized users may only publicly share a subset of all of the landmarks they visited. The fact that Internet volunteered data are biased towards a particularly motivated subset of Internet users is an important limitation of such data. Hence, the resulting analysis might not be representative of the general population. Therefore, before utilizing the proposed methodology in real-world applications for tourist trip planning purpose, the impact of such limitations on the analysis should be considered carefully.

1.6 References

Adrienko, N. & G. Adrienko (2010) Spatial generalisation and aggregation of massive movement data. IEEE Transactions on Visualization and Computer Graphics, 17, 205 - 219.

Ahern, S., M. Naaman, R. Nair & J. H.-I. Yang. 2007. World explorer: visualizing aggregate data from unstructured text in geo-referenced collections. In Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries, 1-10. Vancouver, BC, Canada: ACM.

Andrienko, G., N. Andrienko, P. Bak, S. Kisilevich & D. Keim. 2009. Analysis of community-contributed space-and time-referenced data (example of panoramio photos). In 2009 IEEE Visual Analytics Science and Technology, 213-214. Atlantic City, New Jersey, USA: IEEE Computer Society Press.

Ankerst, M., M. Breunig, H. Kriegel & J. Sander. 1999. OPTICS: ordering points to identify the clustering structure. In 1999 ACM SIGMOD International Conference on Management of Data, 49-60. New York, NY, USA: ACM.

Roy, B. S., G. Das, S. Amer-Yahia & C. Yu. 2011. Interactive itinerary planning. In 2011 IEEE International Conference on Data Engineering, 15-26. Washington, DC, USA: IEEE.

Cao, L., L. Jiebo, A. Gallagher, J. Xin, H. Jiawei & T. S. Huang. 2010. Aworldwide tourism recommendation system based on geotaggedweb photos. In 2010 IEEE International Conference on Acoustics Speech and Signal Processing, 2274-2277. Dallas, Texas, USA: IEEE.

Chekuri, C. & M. Pal. 2005. A recursive greedy algorithm for walks in directed graphs. In 2005 IEEE Symposium on Foundations of Computer Science, 245-253. Washington, DC, USA: IEEE.

Chen, W.-C., A. Battestini, N. Gelfand & V. Setlur. 2009. Visual summaries of popular landmarks from community photo collections. In Proceedings of the 7th ACM International Conference on Multimedia, 789-792. Beijing, China: ACM.

Clements, M., P. Serdyukov, A. P. d. Vries & M. J. T. Reinders. 2010. Using flickr geotags to predict user travel behaviour. In Proceeding of the 33rd international ACM SIGIR Conference on Research and development in information retrieval, 851-852. Geneva, Switzerland: ACM.

Crandall, D. J., L. Backstrom, D. Huttenlocher & J. Kleinberg. 2009. Mapping the world's photos. In Proceedings of the 18th International Conference on World Wide Web, 761-770. Madrid, Spain: ACM.

De Choudhury, M., M. Feldman, S. Amer-Yahia, N. Golbandi, R. Lempel & C. Yu. 2010. Automatic construction of travel itineraries using social breadcrumbs. In Proceedings of the 21st ACM Conference on Hypertext and Hypermedia, 35-44. New York, NY, USA: ACM.

Diaconis, P. (2008) The Markov chain Monte Carlo revolution. Bulletin of the American Mathematical Society, 46, 179-205.

Edelsbrunner, H. & E. Mücke. 1992. Three-dimensional alpha shapes. In Proceedings of the 1992 Workshop on Volume Visualization, 75-82. New York, NY, USA:ACM.

Feo, T. A. & M. G. C. Resende (1995) Greedy randomized adaptive search procedures. Journal of Global Optimization, 6, 109-133.

Fukunaga, K. & L. Hostetler (1975) The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Transactions on Information Theory, 21, 32-40.

Gao, Y., J. Tang, R. Hong, Q. Dai, T.-S. Chua & R. Jain. 2010. W2Go: a travel guidance system by automatic landmark ranking. In Proceedings of the international conference on Multimedia, 123-132. Firenze, Italy: ACM.

Geisberger, R., P. Sanders, D. Schultes & D. Delling (2008) Contraction hierarchies: Faster and simpler hierarchical routing in road networks. Experimental Algorithms, 319-333.

Girardin, F., F. Calabrese, F. D. Fiore, C. Ratti & J. Blat (2008) Digital footprinting: uncovering tourists with user-generated content. IEEE Pervasive Computing, 7, 36-43.

Goldberger, J. & T. Tassa (2008) A hierarchical clustering algorithm based on the Hungarian method. Pattern Recognition Letters, 29, 1632-1638.

Goodchild, M. (2007) Citizens as sensors: the world of volunteered geography. GeoJournal, 69, 211-221.

Grabler, F., M. Agrawala, R. W. Sumner & M. Pauly. 2008. Automatic Generation of Tourist Maps. In Proceedings of ACM SIGGRAPH 2008, 27-3. New York, NY, USA: ACM.

Kopf, J., M. Agrawala, D. Bargeron, D. Salesin & M. F. Cohen. 2010. Automatic Generation of Destination Maps. In Proceedings of ACM SIGGRAPH Asia 2010, 29-6. New York, NY, USA: ACM.

Hagen, K., R. Kramer, M. Hermkes, B. Schumann & P. Mueller (2005) Semantic matching and heuristic search for a dynamic tour guide. Information and Communication Technologies in Tourism 2005, 149-159.

Hoashi, K., T. Uemukai, K. Matsumoto & Y. Takishima. 2009. Constructing a landmark identification system for Geo-tagged photographs based on Web data analysis. In 2009 IEEE International Conference on Multimedia and Expo, 606-609. New York City, NY, USA.

Jaffe, A., M. Naaman, T. Tassa & M. Davis. 2006. Generating summaries and visualization for large collections of geo-referenced photographs. In Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, 89-98. Santa Barbara, California, USA: ACM.

Jankowski, P., N. Andrienko, G. Andrienko & S. Kisilevich (2010) Discovering Landmark Preferences and Movement Patterns from Photo Postings. Transactions in GIS, 14, 833-852.

Kellerer, H., U. Pferschy & D. Pisinger. 2004. Knapsack problems. Springer Verlag.

Kisilevich, S., D. Keim & L. Rokach. 2010a. A novel approach to mining travel sequences using collections of geotagged photos. In Geospatial Thinking, eds. M. Painho, M. Y. Santos & H. Pundt, 163-182. Springer Berlin Heidelberg.

Kisilevich, S., M. Krstajic, D. Keim, N. Andrienko & G. Andrienko. 2010b. Event-based analysis of people's activities and behavior using Flickr and Panoramio geotagged photo collections. In Proceedings of the 2010 14th International Conference Information Visualisation, 289-296. Washington, DC, USA: IV.

Kisilevich, S., F. Mansmann & D. Keim. 2010c. P-DBSCAN: a density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos. In Proceedings of the 1st International Conference and Exhibition on Computing for Geospatial Research & Application, 1-4. Washington, DC: ACM.

Laporte, G. & S. Martello (1990) The selective travelling salesman problem. Discrete applied mathematics, 26, 193-207.

Lu, X., C. Wang, J.-M. Yang, Y. Pang & L. Zhang. 2010. Photo2Trip: generating travel routes from geo-tagged photos for trip planning. In Proceedings of the International Conference on Multimedia, 143-152. Firenze, Italy: ACM.

Lucchese, C., R. Perego, F. Silvestri, H. Vahabi & R. Venturini. 2012. How random walks can help tourism. In Proceedings of the 34th European Conference on Advances in Information Retrieval, 195-206. Berlin, German: ECIR.

Luxen, D. & C. Vetter. 2011. Real-time routing with OpenStreetMap data. In Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 513-516. New York, NY, USA: ACM.

MacQueen, J. 1967. Some methods for classification and analysis of multivariate observations. In Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, 281-297. California, USA: University of California Press.

Montemanni, R. & L. Gambardella (2009) An ant colony system for team orienteering problems with time windows. Foundation Of Computing And Decision Sciences, 34, 287.

Mount, D. & S. Arya. 1997. ANN: A library for approximate nearest neighbor searching. In Proceedings of the 2nd Annual Fall Workshop on Computational Geometry, (available from http://www.cs.umd.edu/~mount/ANN).

Righini, G. & M. Salani (2009) Decremental state space relaxation strategies and initialization heuristics for solving the Orienteering Problem with Time Windows with dynamic programming. Computers & operations research, 36, 1191-1203.

Rigoutsos, I. & A. Floratos (1998) Combinatorial pattern discovery in biological sequences: The TEIRESIAS algorithm. Bioinformatics, 14, 55-67.

Shi, Y., P. Serdyukov, A. Hanjalic & M. Larson. 2011. Personalized landmark recommendation based on geotags from photo sharing sites. In Proceedings of the 5th AAAI Conference on Weblogs and Social Media, 622-625. Barcelona, Spain: ICWSM.

Sint, L. & D. de Champeaux (1977) An improved bidirectional heuristic search algorithm. Journal of the ACM (JACM), 24, 177-191.

Tan, P., M. Steinbach & V. Kumar. 2006. Introduction to data mining. Pearson Education. Addison Wesley. Boston, MA, USA.

Tobler, W. (1987) Experiments in migration mapping by computer. Cartography and Geographic Information Science, 14, 155-163.

Tong, H. & C. Faloutsos. 2006. Center-piece subgraphs: problem definition and fast solutions. In Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 404-413. Philadelphia, PA, USA: ACM.

Canneyt, V. S., S. Schockaert, O. V. Laere & B. Dhoedt. 2012. Time-dependent recommendation of tourist attractions using Flickr. In Proceedings of the Belgian/Netherlands Artificial Intelligence Conference, Ghent, Belgium: BNAIC. (available from http://allserv.kahosl.be/bnaic2011/sites/default/files/bnaic2011_submission_22.pdf)

Vansteenwegen, P., W. Souffriau, G. V. Berghe & D. V. Oudheusden (2011a) The city trip planner: an expert system for tourists. Expert Systems with Applications, 38, 6540-6546.

Vansteenwegen, P., W. Souffriau & D. V. Oudheusden (2011b) The orienteering problem: A survey. European Journal of Operational Research, 209, 1-10.

Vansteenwegen, P., W. Souffriau, G. Vanden Berghe & D. Van Oudheusden (2009) Iterated local search for the team orienteering problem with time windows. Computers & Operations Research, 36, 3281-3290.

Xie, M., L. V. S. Lakshmanan & P. T. Wood. 2011. CompRec-Trip: A composite recommendation system for travel planning. In Proceedings of the 27th International Conference on Data Engineering, 1352-1355. Washington, DC, USA: IEEE.

Yoon, H., Y. Zheng, X. Xie & W. Woo (2010) Smart itinerary recommendation based on user-generated gps trajectories. Ubiquitous Intelligence and Computing, 19-34.

Zheng, Y.-T., Z.-J. Zha & T.-S. Chua (2011a) Mining travel patterns from GPS-tagged photos. ACM Transations on Intelligent System and Technology, 3, 56.

Zheng, Y.-T., Z.-J. Zha & T.-S. Chua (2011b) Research and applications on georeferenced multimedia: a survey. Multimedia Tools and Applications, 51, 77-98.

Zheng, Y.-T., Z. Ming, S. Yang, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, C. Tat-Seng & H. Neven. 2009. Tour the world: Building a web-scale landmark recognition engine. In Proceedings of International

Conference on Computer Vision and Pattern Recognition, Miami, Florida, USA: CVPR.

APPENDIX A

ALGORITHMS

Table 1.1

Algorithm: A k-d tree based OPTICS Algorithm for Massive Datasets

Algorithm 1: k-d tree based OPTICS Algorithm for Massive Datasets

```
FuntionLOPTICS (P, EPS, MIN_Pts)
    For Each point in P
        neighbors= kdtree.getNeighbors(obj, e)
        obj.setCoreDistance(neighbors, e, MinPts)
        OrderFile.write(obj)
        Ifobj.coreDistance != UNDEFINED
            orderSeeds.update(neighbors, obj)
            Forobj in orderSeeds
                neighbors = kdtree.getNeighbors(obj, e)
                obj.setCoreDistance(neighbors, e, MinPts)
                OrderFile.write(obj)
                Ifobj.coreDistance !=UNDEFINED
                    orderSeeds.update(neighbors, obj)
                End If
            EndFor
        End If
    End For
End Function

FunctionOrderSeeds::update(neighbors, centerObj)
    d = centerObj.coreDistance
    For Each unprocessed objin neighbors
        newRdist = max(d, dist(obj, centerObj))
        Ifobj.reachability == UNDEFINED
            obj.reachability = newRdist
            insert(obj, newRdist)
        ElifnewRdist<obj.reachability
            obj.reachability = newRdist
            decrease(obj, newRdist)
        End If
    End For
End Function
```

Table1.2

Algorithm: Extract hierarchical clustering structure from OPTICS reachability plot

Algorithm 2: Automatically extract hierarchical clustering structure from OPTICS reachability plot

```
FuntionExtract_HClusters (P, RD, t)
Clusters = None
While index <len(P)
    If (start = RD[index]*(1-t)) >= RD[index+1] # start of potential steep down area
        index = end_of_down_area
        Steep_down_area.append([start,index])
    Else If (start = RD[index]) <= RD[index+1]*(1-t) # start of potential steep up area
        index = end_of_up_area
        steep_up_area = [start,index]
        For Each D inSteep_down_area
            Ifsuccess_form_cluster(D,steep_up_area) # if D and steep_up_area can form cluster
                Cluster.append(D.start, steep_up_area.end)
            End If
        End For
    End If
End While
root = Node(0,len(p),None) #Node(start,end,parent)
root = generate_dendrogram(root)
Return root
End Function

Functiongenerate_dendrogram(root)
    Range = [root.start, root.end]
    largest_cluster = find_largest_cluster(Clusters, Range)
    center_part = Node(largest_cluseter.start, largest_cluster.end, root)
    root.append_child(generate_dendrogram(center_part))
    Ifroot.start == largest_cluster.start&&root.end == largest_cluster
        Return
    End If
    Left_Range = [root.start, largest_cluster.(start-1)]
    left_clusters = find_clusters(Cluster, Left_Range)
    For Each cluster inleft_clusters
        If cluster in {cluster' in left_clusters}
            left_node = Node(cluster.start,cluster.end,root)
            root.append_child(generate_dendrogram(left_node))
        End If
     End For
    Right_Range = [largest_cluster.end, root.end]
    right_clusters = find_clusters(Cluster, Right_Range)
    ForEach cluster inright_clusters
        If cluster in {cluster' in right_clusters}
            right_node = Node(cluster.start,cluster.end,root)
            root.append_child(generate_dendrogram(left_node))
        End If
    End For
End Function
```

Table1.3

Algorithm: Mining Classic ROI based Travel Routes

Algorithm 3: Mining Classic ROI based Travel Routes

```
FunctionApriori_Mining_Travel_Routes (Travel_Routes, min_support,max_k=None)
    k = 1
    C = find_frequent_k_candidates(k, Travel_Routes,min_support)  # see formula 1.11
    Repeat
        C_Temp= None
        For Each candidate in C
            For Each candidate' in C
                If candidate!=candidate' &&concatenate(candidate,candidate') == True
                    C_Temp[concatenate(candidate,candidate')] = 0
                End If
            End For
        End For
        k = k+1
        For Eachtravel_routeinTravel_Routes
            For Each new_candidate in C_Temp
                If travel_route.contains(candidate)
                    C_Temp[new_candidate] += 1
                End If
            End For
        End For
        C = find_frequent_k_candidates(C_Temp, Travel_Routes, min_support)
    Until C is empty
    Return C
End Function
```

Table1.4

Algorithm: Iterated Local Search heuristic algorithm for multi-stay TOPTW problem

Algorithm 4: *Iterated Local Search heuristic algorithm for multi-stay TOPTW problem*

```
Function MS-TOPTW(start,end,numberOfDays)
    Tour = Initialize_Tour (start, end, numberOfDays)
    StartShake = 1
    ShakeRange = 1
    BestSolution = Tour.total_scores
    WhileNoImprovement< 100:
        // Construct step
        While existing feasible visit
            ForEach spot in Tour:
                Candidates = SearchNearestPOIs(maximum_driving_hours)
                For Eachpoiin Candidates:
                    Calculate BenefitScore of inserting poi in spot
                    Selected_POI, Selected_Spot = GetBestVisit(Candidates, BenifitScores)
                    InsertToTour(Selected_POI, Selected_Spot)
                    For Each POI in Tour:
                        UpdateMaxExtraTime(POI)
                        IfTour.total_score>BestSolution
                            BestSolution = Tour.total_score
                        Else
                            NoImprovement += 1
                        End If
                    End For
                End For
            End For
        End While
        // Shake step
        RemovePOIsFromTour(StartShake, ShakeRange)
        StartShake += ShakeRange
        ShakeRange += 1
        IfStartShake>= min_length(Tour_Segments):
            StartShake = 1
        End If
        IfShakeRange>= length(POIs)/number_visiting_days:
            ShakeRange = 1
        End If
    End While
End Function
```

Essay 2

Introducing Complexity Measures to Trajectory Classification Modeling to Automatically Detect Different Movement Types with Unknown GPS Trajectories

Abstract

The application of trajectory classification to automatically detect movement types of unknown trajectories has been receiving increasing research attention in areas such as video surveillance, traffic management and location-based services. This research applies classic geometric shape-based classification approaches to classify trajectories by utilizing the geometric characteristics of movement to fulfill this task. However, this approach is limited to the geographic context of trajectory data. Classification methods based on movement parameters can overcome this problem but the accuracy of classification depends heavily on selecting appropriate movement features from trajectories. Recent research attempts to extract local movement profiles to improve the accuracy of classification, but is restricted to fixed size trajectories.

To overcome this research challenge, I develop an efficient trajectory classification model based on several movement parameters. This model introduces two new types of complexity measures as new features for classifying movements: (1) the geometric complexity measures of trajectories based on Fractal Dimensions, and (2) structural complexity measures of movement parameters based on Approximate Entropy. I test the feasibility of this proposed

classification model with 800 GPS traces that were shared and manually tagged with four movement types by Internet users on the website Openstreemap.org. The overall 85.4% average accuracy of prediction outperforms the current state-of-the-art trajectory classification models and demonstrates the applicability of this classification model.

2.1 Introduction

In this essay, I develop an approach to efficiently, accurately and automatically detect the movement type of unknown objects from trajectories. This approach addresses several existing research gaps described below and advances the discovery of knowledge and patterns from ubiquitous online trajectory data in a spatiotemporal framework. It is part of a new area of research in spatial behavioral research (see review in section 2.2.1). The emergence of this research is related to the fact that technologies that extract human movement trajectories from various moving object tracking systems are becoming more powerful. Further, the amount of trajectory data is increasing rapidly. Trajectory data can be directly collected with various location-aware devices such as GPS, cell phones (González et al. 2008), WiFi instruments (Torrens 2008) and Bluetooth devices (Eagle and Pentland 2006). Trajectories can also be indirectly extracted from video cameras (Nguyen et al. 2005) or manually recorded using TabletPC (Torrens et al. 2011). Moreover, trajectory data can be reconstructed from some location proxies of physical movement (e.g. using geo-tagged digital photos to reconstruct people's travel paths).

By analyzing very large amounts of trajectory data, scientists can successfully classify trajectories based on different human behavioral types (Dodge et al. 2009), predict the behavioral type of unknown trajectories (Nguyen et al. 2005) and detect abnormal behavior (Makris and Ellis 2002) or critical crowd situations (Johansson et al. 2008). Trajectory classification that can detect the type of movement or behavior (e.g. driving, running or walking) associated with unknown trajectories, is important for deriving knowledge and patterns of movement from trajectory data (Giannotti and Pedreschi 2008). The main task of trajectory classification is to use existing knowledge about the movement behavior to train a model or classifier. Examples of classifiers include the decision-tree model, neural-network, *k*-nearest neighbors, the hidden Markov model (HMM), and the support vector machine (SVM) (see review in section 2.2). Such classifiers serve as an explanatory tool for distinguishing trajectories of different activity types and for predicting, which specific behavior type of any input trajectory data belongs to what predefined categories of behavior.

Trajectory classification plays an important role in many applications of trajectory data analysis and mining. In location-based services, detecting the behavior of moving objects based on their trajectories is a fundamental task of building intelligent systems in smart environments that can determine how to deliver what kind of appropriate services to what types of people (e.g. traffic or gas information to drivers or landmark information to pedestrian tourists). Trajectory classification can also be used to detect abnormal behavior against normal movement patterns from an individual's trajectory dataset in many video surveillance systems. In computer visions, trajectory-based video surveillance systems apply trajectory classification to detect abnormal movement when monitoring traffic, crowds, pedestrians, etc. In artificial intelligence, detecting and identifying unknown moving objects is a basic task of robots and robotic devices for collision free path planning, and trajectory classification could be an efficient approach to achieve this task. In web applications and services, trajectory classification can be used to detect and then automatically tag uploaded raw GPS logs for Internet users. In the behavioral sciences, identifying the movement type of moving objects from raw trajectory data provides fundamental movement information for discovering knowledge of behavioral differences from different moving objects.

Due to the utility and potential wide applicability of trajectory classification, many studies in the behavioral sciences, bioengineering, transportation and video surveillance exist that classify trajectory data to detect the behavior or movement type of moving objects. In general, existing methods can be categorized into two types based on what features of trajectory data are extracted and used to build the classification model: trajectory classification based on (1) geometric shape and (2) movement parameters. The geometric shape approach directly manipulates the spatial characteristics to classify trajectories into one of several predefined categories with similar geometric properties (see review in section 2.2). It is suitable for abnormal behavioral detection from trajectories that were generated by similar moving objects (e.g. trajectories of vehicle in traffic surveillance analysis). However, this type of classification method is limited to the spatial context where all trajectories should be compared in the same geographic region since all predefined trajectory categories are tied to this region. Besides, temporal information has usually been ignored when treating trajectories as two-dimensional line segments since comparing three-dimensional space-time trajectories is computational expensive.

The other type of trajectory classification is based on movement parameters, which are usually descriptive statistics that were extracted from trajectory data to discriminate the differences between movements. Several movement parameters, such as moving speed, turning angles, acceleration etc., have been used in current research as movement features in trajectory classification. Since these movement parameters are not correlated with specific geometric characteristics of trajectories, this type of approach can be applied to any trajectory data regardless of its spatial context. However, the accuracy of classification depends heavily on selecting appropriate movement features from trajectories. With normal movement parameters it is difficult to fully distinguish the differences in movement, especially for similar moving objects. For example, people could run as fast as a slow cyclist (same speed) in the same street (same turning angle). Recent research extracts local movement profiles, such as the amplitude and frequency of movement parameters over time, as new features to discriminate different types of movement in trajectory classification. Other research combines geometric features and movement parameters to classify trajectories. However, these solutions either can only be applied to classify trajectories with fixed duration or cannot generate accurate classification results.

To overcome these research challenges, I develop an efficient approach to automatically —and with high accuracy— detect the movement type of unknown objects from trajectories. In this approach, I extend the movement parameters trajectory classification by introducing two new types of complexity measures as new features to classify movement. Specifically, one type of complexity measure is geometric complexity measured by the Fractal Dimensions of trajectories, and the other is structural complexity measured by Approximate Entropy (ApEn) of the variation in movement parameters. I suggest that ApEn (which provides complexity information about the subtle changes that occur in the structure of sequential movement parameters of trajectories) and Fractal Dimensions (which provide the overall description of geometric complexity) can be used to deal with trajectories with any length and improve the accuracy in trajectory classification.

To demonstrate the utility of this approach, I select 400 GPS traces that have been shared and manually tagged with a specific movement type by Internet users on the website Openstreemap.org. Experiments are conducted to test the feasibility of the two types of movement features introduced in this essay: complexity measures of movement parameters (e.g. ApEn of velocity, turning angle and acceleration sequence data) and complexity measures of geometric shape (e.g. fractal dimensions of trajectory). The performance and accuracy of these trajectory classification models, one with and one without complexity features, is analyzed and then compared using a confusion matrix and receiver operating characteristics (ROC). The overall 85.4% average accuracy of prediction demonstrates the applicability of the method I propose for detecting the movement type of raw trajectory data.

The following sections of this essay are organized as follows: related work on trajectory data mining and trajectory classification are reviewed in section 2.2. The complexity of movement is then introduced in section 2.3 by focusing on two types of complexity measures of movement: ApEn and Fractal

Dimensions. Section 2.4 describes the methodology of classification, which includes data preprocessing, movement feature extraction and dimensional reduction, building the classification model, applying this model in movement type detecting, analyzing the results and comparing them with other approaches. In section 2.5, two experiments are conducted to validate the performance trajectory classification model. The results are compared to evaluate the effects of the two proposed complexity based movement features. Section 2.6 presents a brief summary and outlines limitations and future research opportunities in this area.

2.2 Related Work

Discovering knowledge from trajectory data has gained much attention in recent years. In Ashbrook and Starner (2003), a Markov model was employed to learn the traveling patterns of new residents in Zürich, Switzerland from trajectory data. The model was trained using a region-to-region transition matrix where each region represents a cluster of a group of fixed points on trajectories. Mamoulis et al. (2004) proposed a learning based approach to mine the periodic commuting patterns from trajectory data. They clustered the trajectory data into different regions, and detected the dependences between regions through association rules. Giannotti et al. (2007) developed a T-Pattern to describe aggregate travel patterns in urban areas by analyzing individual-level trajectory data. The T-Patterns, which are flows between regions of interest (ROIs), were mined from Point of Interests extracted from trajectory data. Zheng et al. (2009) proposed an inference model based on HITS(Hypertext Induced Topic Search) to mine classic travel sequences in Beijing, China. This model was trained with the trajectories of users' travel experiences. Adrienko and Adrienko (2010) provided a novel approach to mine the travel patterns of residents in Milan, Italy and visualized them as aggregate flows between areas. Their study area was first divided into several regions as Voronoi tessellations, and travel patterns were then mined from trajectory data, which were segmented into a sequence of connected regions. These movement patterns provide relevant information for traffic management, urban design, local facilities design, migration, and crime analysis.

Trajectory classification, which detects individuals or groups with similar or divergent movement behaviors from trajectories, is an important task for providing fundamental information for today's trajectory data analysis and mining (Dodge et al. 2008). Further, trajectory classification has been applied in many fields. For example, it can be used to detect abnormal pedestrian behavior in pedestrian video surveillance systems (Niu et al. 2004), abnormally moving vehicles in traffic video surveillance systems (Fu et al. 2005), vessel types for fishery control, pollution control and border control from satellite images (Lee et al. 2008), and the theft of mobile devices (Yazji et al. 2011). The trajectory classification approaches can be divided into two types based on what features of trajectory data are used for classification: the first approach is based on geometric shapes and the second one on movement parameters.

2.2.1 Trajectory Classification based on Geometric Shape

Intuitively, in trajectory classification based on geometric shape, whole trajectory or partitioned trajectory segments can be used to compare the geometric similarity between trajectories for further classification. Therefore, the main task of this approach is to compute the geometric similarity between trajectories (see chapter 10 in Giannotti and Pedreschi 2007). Many approaches were developed to compute overall visual distance or similarity between trajectories, such as average or perpendicular Euclidean distance (Froehlich and Krumm 2008), Fréchet distance (Buchin et al. 20011) and Longest Common Subsequence (Lin and Shim 1995). Once the distances between trajectories are calculated, generic clustering algorithms in data mining, such as k-means, density based models or the Gaussian mixture model can be applied to find possible clusters as predefined categories for further trajectory classification.

However, such methods are usually limited to the high complexity of geometry computation and inconsistency of spatial context in trajectory data. Besides, since trajectory is a physical representation of an individual’s spatial behavior that changes simultaneously in space and time, geometric shape methods always ignore the temporal constraints of trajectories for simplicity's sake. For example, trajectories along the same path might have reverse moving directions that might indicate two different moving patterns. Even though researchers developed some algorithms for comparing trajectories that take both space and time into account (Vlachos et al. 2003), they are still computationally expensive and cannot be applied efficiently to large scale trajectory data.

To solve such issues and take the temporal variations in trajectories into account, Markov models such as hidden Markov models (HMMs) and hierarchical hidden Markov models (HHMMs) can be used to model the movement process for classifying each movement class. In these Markov models, trajectory data need to be segmented or partitioned to several connected sub-trajectories. Then, each segmented sub-trajectory can be treated as a state and movement can be represented as the transitions between states. The state transition probabilities in the Markov models can be learned from training trajectory data. When modeling a new trajectory, the Markov model with the highest likelihood is used to classify the new input. However, like classic geometric shape based approaches, the Markov models are still restricted to be applied on trajectories in the same geographic region because all predefined trajectory categories are tied to this region.

In Bashir, Khokhar et al. (2007), a temporal independent Gaussian Mixture Model (GMM) and a temporal dependent hidden Markov model (HMM) were created and compared for classifying trajectory data extracted from people's signatures and a sports video dataset. The trajectories were firstly partitioned into trajectory segments at points of change in curvature and represented as a sequence of transitions between sub-trajectories. In GMM-based classification, for each trajectory class, the probability density function (PDF) is estimated with GMM using the sub-trajectories in this class. The classification of new trajectories can be performed by computing the likelihood for each GMM that has been trained from trajectory samples of each predefined category. The category with the highest likelihood will be assigned to the input trajectory. In the HMM-based approach, a sub-trajectory was used to model the state of the HMM by using a mixture of Gaussians. For each class, a HMM was trained with known sub-trajectories. For a new trajectory, the HMM with the highest likelihood is used to describe its class. Their experiments suggest that HMM-based trajectory classification gains higher accuracy than GMM.

To predict the behavioral type of unknown trajectories extracted from video surveillance, Nguyen et al. (2005) presented a hierarchical hidden Markov model (HHMM) that learns the transition rules of human behavior from sequential trajectory data. In their approach, the study area was divided into several regions, and then movement trajectories were represented as a connected sequence of regions. HHMs of high-level behaviors (e.g. short meal, have snack etc.) and their sub-HHMs of low-level movement (transitions between regions) were organized hierarchically in a lattice-like structure for building a HHMM. In their model, the internal and external factors that stimulate people's behaviors are treated as hidden factors that force people to change their behavior.

2.2.2 Trajectory Classification based on Movement Parameters

Another type of trajectory classification is based on movement parameters. In most cases, geometric shape-based descriptors cannot fully discriminate the differences between trajectories due to the complexity of geometry shapes and their spatial context. For example, a straight trajectory recorded by people walking along a crowded street in New York has similar geometric characteristics as a straight trajectory recorded by people drive along the same street, and these are two different movement types of trajectory. This could be tackled by using movement parameters (e.g.velocity, acceleration, turning angle, etc.) that are extracted from trajectories to discriminate the different movement types of trajectory. Finding good features from movement parameters to build the classification model is key for the accuracy and robustness of trajectory classification.

Niu, Long et al. (2004) used statistical movement properties that are extracted from trajectories to detect group movement behavior in video surveillance systems, such as following, following-and-gaining and stalking, of pedestrians. The movement properties are relative moving position and relative moving velocity of one pedestrian vs. a nearby pedestrian. The linear regression models of these movement properties against time are then evaluated, and the characteristics of the best-fit regression lines, such as slope, intercept and residual error, are used as features for building a SVM-based classifier from training data. Their experimental results show better classification results than complicated Markov models, such as HMM and coupled hidden Markov models.

For efficient trajectory classification, Dodge et al. (2009) proposed a trajectory segmentation and feature extraction method for detecting movement type of moving objects from trajectory data. They used an analytical approach to extract descriptive statistics, including speed, acceleration turning angle, straightness, etc., as global movement parameters from trajectory data. To extract more detailed features of movement, they decomposed trajectories into segments at equal time intervals and measured the amplitude (e.g. using deviation from the median) and frequency variations (e.g. using sinuosity) with regard to global movement parameters over time for each segment. The deviation and sinuosity of movement parameters are then categorized into four predefined low-high deviation-sinuosity groups. The statistics of four groups extracted from trajectories were used as local movement features. For trajectory classification, they first used principal component analysis (PCA) to reduce the dimension of the movement features, and then trained a SVM classifier from training data to classify trajectories with equal time duration into categories such as pedestrian, bicycle, car and motorcycle with a 82% accuracy of multi-label classification. However, this is limited to a prerequisite that all trajectory samples need to have the same time duration.

Based on their previous work, Dodge et al. (2012) proposed to assign one of the predefined movement parameter classes(MPC) to each trajectory segment based on different value ranges of its local movement features (e.g. deviation and sinuosity) and symbolically represent each trajectory as a sequence of class labels. They measured the similarity between different sequences by using a so-called normalized weighted edit distance (NWED). This method can be applied to trajectories with different length, which overcomes the deficiency of their previous approach. They applied their approach to a similarity-based trajectory clustering task using North Atlantic Hurricane trajectory data and GPS traces of couriers in London.

There are also some researchers tried to utilize both geometric characteristics and movement parameters of trajectory to identify different movement types. For example, to detect abnormal movements among vehicles in real-time traffic videos, Fu et al. (2005) developed a hierarchical clustering framework that uses a spectral clustering algorithm to identify normal trajectory clusters. A set of decision rules is then defined to detect abnormal vehicles by comparing the visual similarity, spatial and velocity constrains of their moving trajectories with a template trajectory that is extracted in each cluster. All trajectory data were resampled at equal space intervals and represented as line segments. The average distance between corresponding comparable segments on two trajectories is used as visual features to compare the similarity between trajectories regardless of differences in length of trajectories.

2.3 Complexity of Movement

A trajectory is a path of connected geometric line segments that can be treated as a type of time series data. Therefore, many existing methods in computational geometry and time series analysis have been borrowed for trajectory analysis and classification. For example, several geometry-based approaches are developed to compare geometric similarities between trajectories (see review in section 2.2.1). Further, trajectory classification research borrowed methods from time series analysis to compare similarities between time series data —for instance, to detect change in a time series for trajectory segmentation or to extract features of time series (e.g. amplitude, frequency and variations). Approaches for investigating time series data have also been used in trajectory classification, such as Markov models or dynamic time warping. However, to the best of this author's knowledge, none of the existing research introduces complexity measures, which have been widely studied in fractal theory (Batty 1985) and time series research (Feldman and Crutchfield 1998) to describe the characteristics of movement for trajectory classification. In this essay, I seek to demonstrate that complexity measures of trajectories can provide new and discriminative features of movement for trajectory classification. I introduce two types of complexity measures for trajectories: a geometric complexity measure using Fractal Dimensions and a structural complexity measure of movement parameters using Approximate Entropy.

2.3.1 Geometric Complexity of Movement and Fractal Dimension

When trajectories are visually plotted in two-dimensional space (see upper figure 2.1), the fundamental feature of trajectories is their geometric shape. Much existing research focuses on directly comparing the geometric shape between trajectories to identify similar movements with several limitations (see review in section 2.2.2). In fact, the geometric shape of a trajectory itself can tell us the characteristics of movement via its geometric complexity measure. For example, people walking through a crowded street block may generate a trajectory full of angles and turns by avoiding collisions or visiting random places, while a car that drives through the same street will create a straight trajectory. The geometric complexity of these two movement trajectories is significantly different: the trajectory of pedestrians in a crowded environment is more complex geometrically than the trajectory of vehicles. I argue that such geometric complexity of trajectories can be used as a discriminative feature to describe movement.

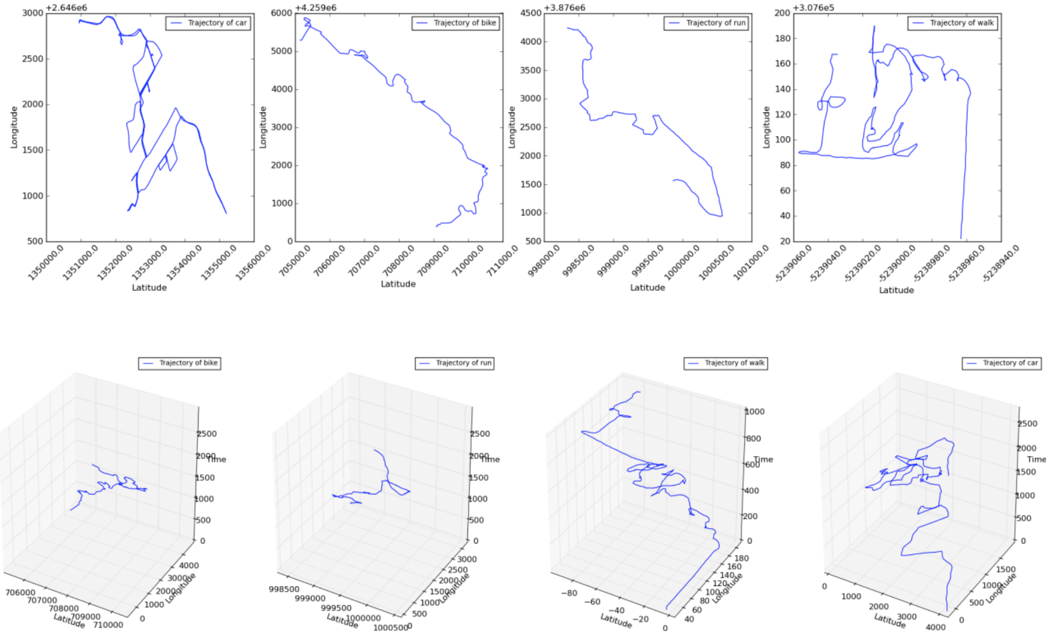


Figure 2.1: 4 different randomly selected GPS trajectories (car, bike, run, walk) in 2D (upper) and in 3D (lower, with vertical axis representing time)

To measure the geometric complexity of trajectories, I introduce the Fractal Dimension (FD) as a geometric complexity-based feature for trajectory classification. Fractal dimension is used to measure the tortuosity of two-dimensional trajectories (Mandelbrot 1967, Nams 2005). It has been used to analyze the trajectories of animals to study their movement patterns and habits (Fritz et al. 2003) and to analyze the structure of trajectories of pedestrians (Nara and Torrens 2007) to compare the visual similarity between trajectories (Torrens et al. 2011). The FD value of a trajectory ranges from 1, which refers to a straight line, to 2, which means a trajectory whose tortuosity occupies a whole plane. It is derived from the linear relationship between the logarithm of total distance (D) and the logarithm of the inverse of the currently employed measuring scale (S) based on the knowledge that the total length is highly dependent on the scale adopted (Nams 2005), as follows:

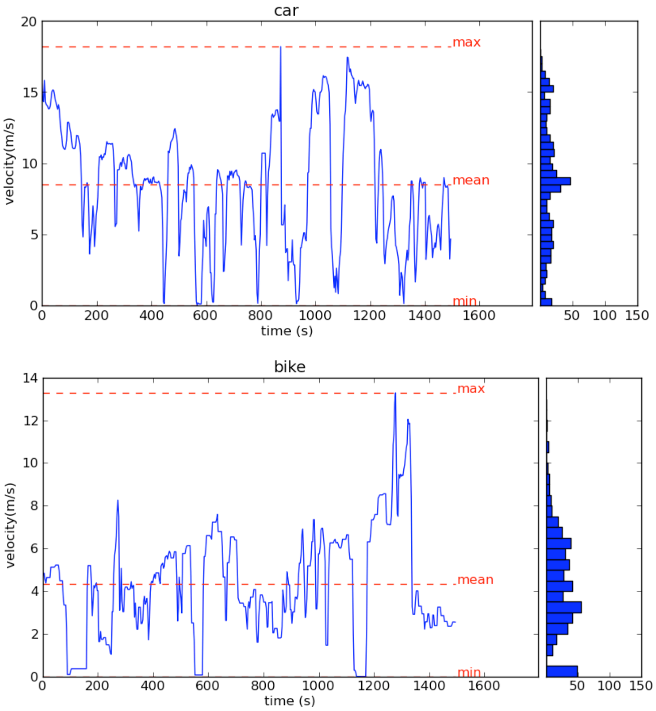
$$\log(D_i) = \beta + \alpha \log 1/S_i \ , i \epsilon [1,2, \dots, n] \tag{2.1}$$

where n represents the number of different scales employed to calculate the total distance of a trajectory. A regression model can be constructed from the (D_i,S_i) pairs, and the FD value is then calculated as $(1 + \alpha)$. One problem that may impact the precision of the FD value is a possible underestimation or truncation of the path length through different measuring scales. For the purpose of improving precision, Nams (2005) proposed a so-called FMean method that computes an FD value twice by starting to measure total distance from two ends of a trajectory whereby the mean FD value is used as FMean. In this research, FMean will be used as a movement feature that describes the geometric complexity of trajectories.

2.3.2 Structual Complexity of Movement Parameters and Approximate Entropy

Besides the geometrical shape, the global characteristics of trajectories can be described through some movement parameters, such as average velocity, acceleration, turning angle, straightness index etc. (see review in section 2.2.2). These descriptors, at a given scale, can differentiate a variety of behaviors. For example, in most cases, people are running with higher moving speed than walking; driving a car will be associated with a much higher acceleration than riding a bike; the turning angle of a vehicle will be smaller than that of a pedestrian. These differences can be seen in table 2.1, where the basic descriptive statistics of several global movement parameters were computed empirically from four different types of movement trajectories (walk, run, ride bicycle and drive car) that are randomly selected from experimental data (see section 2.5). However, in some cases these descriptive statistics would not be accurate: some people might run very slowly while others might walk very fast, or in a race, a bicycle could reach a fairly high speed that is faster than a slowly driven car. Therefore, additional features that can distinguish different movement parameters of trajectories are needed for a successful classification task.

By plotting the sequential data of these movement parameters (velocity, acceleration and turning angle) against time, we can see that obvious structural differences of different types of movement exist: different behavior exhibits different amplitude and frequency variations of its movement parameters along the time axis (see figure 2.2). In this example, for a velocity-time sequence, running behavior has the relatively highest frequency and median amplitude; walking behavior exhibits the relatively median frequency and lowest amplitude; driving a car is associated with the relatively lowest frequency and highest amplitude; and riding a bike has the relatively median frequency and amplitude. Many approaches to analyzing movement were designed to quantitatively and statistically analyze these time-series data by checking the shifts in mean levels, variability, and the autocorrelation structure. For example, Nams (1996) proposed a so-called VFractal to measure the fractal dimension of the turning angles series to evaluate the self-similarity (autocorrelation) of movement. Dodge et al. (2009) measured the deviation and sinuosity of these sequential movement parameters and categorized them into four predefined low-high deviation-sinuosity groups as the local movement features for trajectory classification. However, such methods that only take into account the aggregate amount of randomness of the serial data may ignore the subtle changes that happened in the structure of sequential data (Pincus 2008).



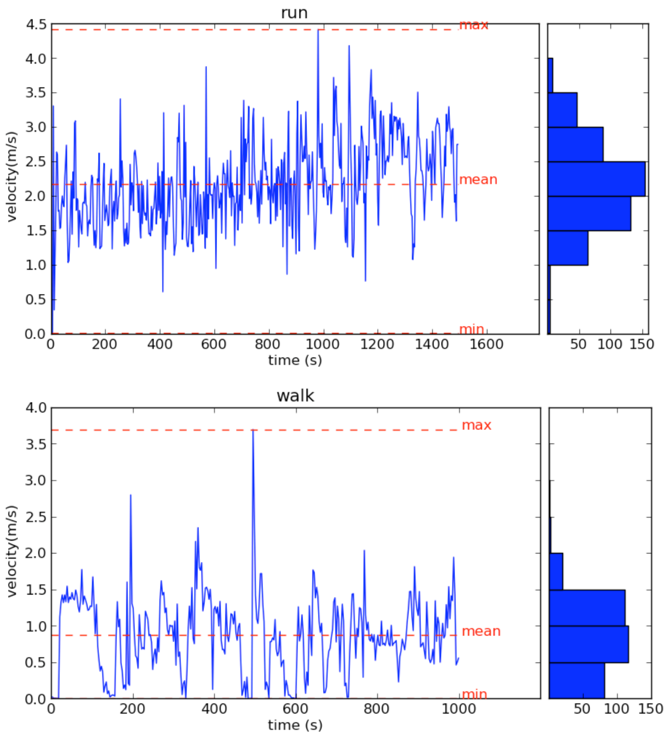


Figure 2.2: Plots of velocity against time for: car, bike, run, and walk (from top to bottom) of 4 randomly selected trajectories with 4 different movement types.

To address this problem, trajectories were treated as time series data and the structural complexity measurement Approximate Entropy (ApEn) was introduced as a measure of irregularity of sequential data in time series analysis. ApEn is rooted in information entropy developed by (Shannon 1948). It is used to quantify the concept of changing complexity, and it has been widely applied in time series data analysis in finance, biology, complexity, and other fields (Pincus 2008, Pincus 1991). The ApEn value varies inversely with complex and irregularity of sequential data. It measures if a structure or pattern of change exists in sequential data. A higher ApEn value suggests that the sequential data is a random series, while a smaller value implies less complexity and more regularity (predictable pattern) in the sequential data. Therefore, this research applied the ApEn to measure the structure of sequential data with local movement parameters.

ApEn values reflect the likelihood of how often "similar" patterns of observations exist in time series data. Sequential data that contains many repetitive patterns (e.g. highly structural and less informative) have a relatively small ApEn value, while a less predictable process (e.g. with complex or random structure) has a higher ApEn value. Given time sequence data S_N, which has N continuous observations, I denote a subsequence of m observations at location i, $i \in [1, N]$, is a pattern $p_m(i)$. If the difference between two patterns $p_m(i)$ and $p_m(j)$ is less than a predefined criterion r, we can conclude that these two patterns are similar. The approximate entropy value $ApEn(S_N, m, r)$ can be computed with the following equation:[7]

$$ApEn(S_N, m, r) = ln\left[\frac{C_m(r)}{C_{m+1}(r)}\right] \tag{2.2}$$

where m specifies the pattern length, r defines the criterion of similarity between patterns, and $C_m(r)$ is the prevalence of repetitive patterns of length m in S_N, which can be computed as:

$$C_m(r) = \sum_{i=1}^{N-m+1} n_{im}(r)/(N-m+1)^2 \tag{2.3}$$

where $n_{im}(r)$ is the frequency count of patterns in P_m that are similar to $p_m(i)$. For a fixed number of N observations, large m will generate fewer patterns to

[7] See http://physionet.org/physiotools/ApEn/

measure the ApEn value than small m. As noted by the author of ApEn in (Pincus 1991), a small m (especially m=2) can distinguish a wide variety of systems, such as deterministic systems, chaotic system stochastic and mixed system, with relatively fewer points. For similarity criterion r, smaller r usually leads poor conditional probability with more similar patterns been identified, while larger r usually ignore detailed system information with less patterns been detected. As suggested by author in (Pincus 1991), choices of r ranging from 0.1 to 0.2 standard deviation of the sequence data S_N can avoid a significant contribution from noise in an ApEn calculation.

2.4 Methods

To detect the movement type of an unknown object from trajectories, I develop a trajectory classification framework based on movement parameters. I introduce two new movement features that represent a trajectory's geometric complexity and structural complexity of movement parameters. First, all trajectory data will be preprocessed in this framework by removing noise and outliers and resampling with a uniform time interval. Then, general movement features (e.g. velocity, turning angle, acceleration and straightness) and complexity-based movement features are extracted from trajectories (e.g. fractal dimension (FMean) and ApEn measures with regard to general movement features). Correlation analysis is then applied to study potential interrelationships between movement features. To reduce the dimensions of movement feature space, principle component analysis (PCA) is used to select a subset of uncorrelated features as principal components. The features and corresponding movement types are then used to train a classifier for trajectory classification. Different classifiers have been compared and the one with highest accuracy is selected to use. This classifier can be used to predict the movement type of an unknown trajectory.

2.4.1 Data Preprocessing

Before data preprocessing, I establish some definitions for trajectory data that can be recorded through location-aware devices (e.g. GPS) at a certain sampling interval or instantaneously through user intervention: a trajectory is the path of a moving object and it can be composed of a set of quasi-linear segments where the points $P = \{p_0, p_1, \dots, p_n\}$ are attributed spatial and temporal information, e.g. $p_i = \{Lat_i, Lon_i, Time_i\}, i\epsilon[0, n]$. Based on this, a trajectory can be represented as follows:

$$Trajectory = p_0 \xrightarrow{\Delta t_0} p_1 \xrightarrow{\Delta t_1} p_2 \xrightarrow{\Delta t_2} \dots \xrightarrow{\Delta t_{n-1}} p_n \qquad (2.4)$$

where $p_i \epsilon P$, $\Delta t_{i+1} = t_{i+1} - t_i$ and $i\epsilon[1,n]$. The total cost in time is $T = \sum_{i=0}^{n-1} t_i$ and the approximate total length of the trajectory is

$$D = \sum_{i=0}^{n-1} distance(p_i, p_{i+1}) \qquad (2.5)$$

. If $\Delta t_0 = \Delta t_1 = \cdots = \Delta t_n$, this trajectory has a fixed sampling interval.

Usually, real-world trajectory data may not have been recorded at the same sampling rate and there may be some noise, such as incorrect locations that were recorded when location-aware devices (e.g. GPS) lost signals or were impacted by ionospheric and tropospheric errors in the trajectory data (Hoffmann-Wellenhof et al. 2001). The different sampling rate should be standardized for generating comparable Fractal Dimension and ApEn values. First, all trajectories that were recorded using latitude and longitude are simply transformed to a planar coordinate system with meters as the unit. To preprocess trajectories to contain the same fixed sampling interval, I then adopted a linear interpolation approach to resample trajectories at a fixed time interval. To check the noise in trajectory data, I applied a simple rule that moving velocity at each original point on the trajectory should be less than a predefined maximum velocity in the resample stage. The noise point will be simply removed once detected before resampling. If more than 10 noise points are detected, this trajectory will be ignored.

2.4.2 Feature extraction and selection

To find a feature set with distinguishable features to better evaluate the characteristics of movement compared to existing work, I propose a movement feature set that can be retrieved from trajectories for a classification task. I introduce two new types of movement features: geometric complexity of movement and structural complexity of the variation of movement parameters. In this movement feature set, classic general descriptive statistics of movement parameters, which include the mean, standard deviation and skewness of moving speed, acceleration, turning angle and straightness index, are extracted from trajectories as movement features. At each sampling point p_i along trajectory$j$ with total n sampling points, these movement parameters can be calculated as follows:

$$Speed_{p_i,t_i} = distance(p_{i+1}, p_i)/\Delta t_i, \tag{2.6}$$

$$Acceleration_{p_i,t_i} = Speed(p_{i+1}, p_i)/\Delta t_i, \tag{2.7}$$

$$TurningAngle_{p_i,t_i} = \angle\theta(p_{i-1} \rightarrow p_i, p_i \rightarrow p_{i+1}), \tag{2.8}$$

$$Straightness_{p_i,t_i} = \frac{distance(p_{i-1},p_i)+distance(p_i,p_{i+1})}{distance(p_{i-1},p_{i+1})}. \tag{2.9}$$

Instant moving speed is calculated as the rate of location change from the previous time step. Acceleration is calculated as the rate of speed change from the previous time step. Turning angle is calculated as the direction of the movement with regard to the previous and next time steps (see figure 2.3).The straightness index is calculated as the ratio of the length of two consecutive trajectory segments and the displacement from an overall start point to end point of these two segments.

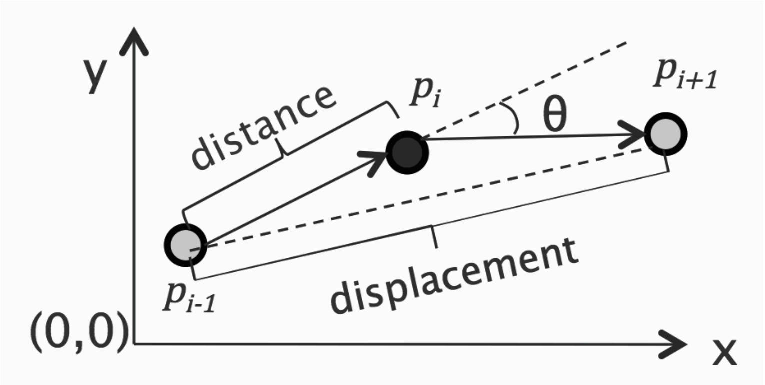


Figure 2.3: Illustration of computing general movement parameters such as: moving speed, turning angle, displacement etc.

To test the potential interrelationships between movement parameters, the Spearman correlation coefficient and the *p*-value for testing non-correlation are adopted. The main reason of selecting Spearman correlation is because it does not assume a normal distribution of the variables. It is a nonparametric measure of the linear relationship between two variables, and can be used to test the direction and strength of the relationship between variables (Chatfield 2004). The correlation measure varies between -1 (strong negative correlation) and +1 (strong positive correlation). Value 0 means no correlation between two variables. Strong correlation between movement parameters implies that some parameters may be redundant and need to be removed.

The new type of movement feature is the complexity measure of a trajectory. I introduce geometric complexity of a trajectory and structural complexity of movement parameters of a trajectory. The Fractal Dimension is calculated to describe the geometric complexity of a trajectory. To obtain better precision, FD is computed twice by measuring the distance of a trajectory in opposite directions using different scales. Further, the mean value (FMean) is used as a movement feature (see section 2.3.1). ApEn (see section 2.3.2) is calculated for each movement parameter (e.g. speed, turning angle, acceleration and straightness) to describe how the structural complexity of movement parameter varies over time. To capture all subtle changes that occurred in the structure of sequential data, the ApEn value of each movement parameter is measured at every sample point beginning from a ½ trajectory. To calculate ApEn values that can distinguish different movement types significantly, the parameters of ApEn are defined as m=2 and r=0.2*standard_devation(S_N) following by the explanation in section 3.2. Then, the mean, standard deviation and skewness of ApEn values are adopted as movement features.

However, correlations may exist in the above movement features extracted from trajectories since some of these features describe similar characteristics of movement. For example, the fractal dimension and straightness index are both used to measure the geometric characteristics of a trajectory. Meanwhile, correlation analysis is difficult to apply to identify and reduce duplicate features due to the relative large number of features. A traditional and efficient approach, principal component analysis (PCA), is employed to reduce the dimensions of feature space by using an orthogonal transformation to reduce a set of possible correlated features to a smaller set of values of uncorrelated synthetic features (Smith 2002). These uncorrelated features are called principal components that contain the most important information of the original features.

2.4.3 Classification Model

After the process of dimension reduction, the final feature set and movement type of trajectories will be used for trajectory classification. Classification is the task that assigns objects to one of a number of predefined class labels based on the feature set of objects. The function that maps each feature set to a discrete class label is called a classification model or a classifier. The classifier is normally trained and evaluated by applying a learning algorithm to identify a model that best fits the relationship between features and classes from training data. Many classifiers have been developed in data mining and machine learning for the classification task, and many of them have been successfully applied in trajectory classification, such as rule-based classifiers, decision trees, SVM, Markov models (e.g. HMM and HHMM) and Bayesian models (see review in section 2.2).

In this essay, a cross-model comparison using different classification models, such as SVM, decision tree, *k*-nearest neighbor (KNN) classification, linear model, naïve Bayes and Gaussian Mixture model (GMM), is applied to select a suitable classification model. As a result, I adopted the SVM as the classifier for trajectory classification since it achieves the highest accuracy in prediction test and has been successfully applied in many applications (Bishop 2006). SVM is also robust for high-dimensional and linearly or non-linearly separable data. It finds maximal margin hyperplanes as decision boundaries to separate input features with different class labels in a multidimensional space. A subset of the training data, the support vectors is used to represent such decision boundaries. For non-linearly separable data, SVM applies a set of kernels, such as linear, polynomial, radial basis function (RBF) and sigmoid kernels, to mathematically map input features to a linearly separable space. After the SVM classifier has been trained, it can be used to classify trajectories into predefined categories and detect the movement type of a trajectory by assigning a class label to it.

2.4.4 Trajectory Classification, Prediction and Evaluation

Trajectory classification can be applied in two major tasks. One is distinguishing trajectories of different movement types based on the movement features that are extracted from trajectories. It can be used to process trajectory data for further analysis and data mining work. The other one, which is the main focus in this essay, is predicting the movement type (class label) of unknown trajectories based on the retrieved movement features from trajectory data. If there are only two predefined movement types that need to be detected, such as walk vs. run, the corresponding classification model is called a binary classifier. Most classification techniques are suited for predicting trajectories with binary categories. When there are more than two movement types in classification, the corresponding classification model is called a multi-class classifier or multinomial classifier. There are some feasible solutions that apply binary classifiers to solve this *k*-classes classification problem, such as a “one-against-one” or “one-against-rest” strategy (Tan et al. 2002). In a “one-against-rest” strategy, *k* binary classifiers will be trained first and then work together as a multi-class classifier. The unknown trajectory will be classified *k* times using these *k* classifiers and generate *k* probability values to indicate whether or not it belongs to each one of *k* movement types. The movement type with the highest classification probability (“one-against-rest”) will be assigned to the unknown trajectory.

To select a suitable classification model in the experiments, I apply a cross-model comparison using different classification models, which include SVM, decision tree, *k*-nearest neighbor (KNN) classification, linear model, naïve

Bayes and Gaussian Mixture model (GMM). According to the results, the support vector machine, which achieves the highest prediction precision, is selected to fulfill this classification task. To avoid the over-fitting problem and improve the estimation of the classification performance, I apply cross-validation to evaluate the classifier. Specifically, a *k*-fold cross-validation method is used. This method divides the sample data into *k* equal-sized groups, from which one group of samples is chosen for testing and the rest of the data are used for training at each run. Then, the overall error equals the sum of errors for all *k* runs.

To evaluate the performance of the classification model, I use classification performance metrics such as accuracy (the ratio of the number of correct predictions to the total number of predictions) and error rate (the ratio of the number of wrong predictions to the total number of predictions). Further, I adopt a receiver operating characteristic (ROC) curve to display the tradeoff between true positive rate (TPR equals the ratio of the number of true positive cases to the sum of true and false positive cases) and false positive rate (FPR equals the ratio of the number of false positive cases to the sum of true and false positive cases) (Hanley and McNeil 1983). The area under the ROC curve (AUC) can be used to evaluate if the model is accurate (with an AUC value close to 1) or inaccurate (with an AUC value close to 0.5), or compare which model performs well (with a large AUC value).

2.5 Experiments and Results

2.5.1 Data Collection

In this research, I retrieved 7,010 GPS tracks that were shared by 478 Internet users in GPS exchange format (GPX) from the website Openstreetmap.org. GPX is an open file format that uses the XML schema to describe waypoints, tracks and routes. In the experiments, all trajectories were extracted from GPX files by using a Python program. Usually, when Internet users upload and share their GPS tracks on a website, most of them also tag their GPS traces with some text descriptions, such as the movement type, date or other relevant information about the GPS traces. These meta-data were also collected with the GPX data at the same time. By using these metadata, I developed another Python program to extract trajectory samples that have metadata that match my four predefined movement categories. The GPS traces, which were tagged with more than one movement type (e.g. GPS trace of commuting or traveling usually contains walking and driving car), will be ignored. This program also detects and deletes invalid trajectory data, such as empty GPX files or too short GPS traces (less than 5 minutes). After cleaning the data, 400 valid trajectories were randomly selected so that each movement category contains 100 trajectories. These trajectories will further be used as training and testing data in the experiments.

The main purpose of the experiments is to build a classifier from already known trajectory data to predict movement types of unknown trajectories from four predefined movement categories: walk, run, ride bicycle (bike) and drive vehicle (car). This trajectory classification approach can be widely applied. For example, for GPS data shared on websites, this method can be used to help Internet users automatically tag their uploaded GPS traces with correct movement labels. It can also be used to analyze the trajectory database to study movement behavior and patterns. Finally, it can also be applied to benefits from other applications that need to identify people's movement type from trajectory data, such as location-based services, video surveillance, and traffic management.

2.5.2 Data Preprocessing, Feature Extraction and Selection

In the data preprocessing stage, outliers in each trajectory, such as points with zero latitude and zero longitude or with moving speed larger than 100 meters per second were removed. Each trajectory is then re-sampled at a fixed time interval (3 seconds) through a linear interpolation approach. As proposed in section 2.4.2, the movement features include (1) general features such as the mean, standard deviation and skewness of movement parameters (speed, acceleration, turning angle and straightness index) and (2) complexity features such as the FMean measure of trajectories, and 3) the mean, standard deviation and skewness of ApEn measures of the variation of movement parameters. The descriptive statistics of movement parameters of four movement types are calculated from sample trajectories and are shown in table 2.1 and table 2.2.

Table 2.1

Descriptive Statistics and Structural Complexity Measures of 4 movement parameters (Speed (a), Acceleration(b), Turning Angle(c), Straightness(d)) for 4 Randomly Selected Trajectories (Car, Bike, Run, Walk)

	Speed (meters/second)			ApEn of Speed		
	Mean	Stddev	Skewness	Mean	Stddev	Skewness
Car	8.503	4.262	-0.001	0.371	0.033	-0.161
Bike	4.376	2.054	-0.843	0.467	0.026	-0.445
Run	2.176	0.634	0.080	0.972	0.131	-0.342
Walk	0.799	0.486	-0.146	0.549	0.101	-0.685

(a)

	Acceleration (meters/second^2)			ApEn of Acceleration		
	Mean	Stddev	Skewness	Mean	Stddev	Skewness
Car	-0.007	0.468	-2.079	0.827	0.022	-0.971
Bike	9.763	9.763	-0.095	0.662	0.037	-0.468
Run	0.001	0.212	0.256	1.077	0.106	-0.677
Walk	3.764	0.114	0.045	0.703	0.083	-0.183

(b)

	Turning Angle (-3.14-3.14 degrees)			ApEn of Turning Angle		
	Mean	Stddev	Skewness	Mean	Stddev	Skewness
Car	-0.018	0.491	-1.26	0.401	0.030	0.146
Bike	-0.006	0.474	0.039	0.424	0.045	0.069
Run	-0.007	0.337	-0.870	1.026	0.162	-0.464
Walk	-0.059	0.786	0.001	0.437	0.072	0.213

(c)

	Straightness			ApEn of Straightness		
	Mean	Stddev	Skewness	Mean	Stddev	Skewness
Car	1.031	0.177	9.419	0.131	0.017	-0.347
Bike	1.026	0.191	17.447	0.175	0.034	0.174
Run	1.015	0.064	17.118	0.313	0.022	0.249
Walk	1.086	0.314	7.004	0.338	0.024	-0.343

(d)

Table 2.2

Geometric Complexity Measures (Fractal Dimensions) of 4 Randomly Selected Trajectories with 4 Different Movement Types (Car, Bike, Run, Walk)

	Car	Bike	Run	Walk
FMean of Fractal Dimensions	1.089	1.083	1.076	1.122

The Spearman correlation coefficient is computed to examine the potential interrelationships between movement parameters. The results are shown in Table 2.3. From the results, we can see that there is a slight positive correlation between "speed" and "acceleration" in two movement types ("car" and "bike"). Therefore, all four movement parameters will be kept for the next stage.

After the correlation analysis, a total of 25 movement features are derived from each trajectory: 12 general movement features (mean, standard deviation and skewness of speed, turning angle, acceleration and straightness index) and 13 complexity movement features (FMean of trajectory, mean, standard deviation and skewness of the ApEn measure of speed, turning angle, acceleration and straightness index curves). Then PCA is applied for dimensional reduction of the above movement features by transforming input features to uncorrelated linear combinations. As a result, the original feature set is reduced to 10 principal components, which together contribute 90% of the original information. The new feature set is then used for the final trajectory classification.

Table 2.3

Correlation Coefficients between Movement Parameters of 4 Movement Types

Correlation	Car	Bike	Run	Walk
Speed-Acceleration	0.576	0.406	0.254	0.056
Speed-TurningAngle	0.128	0.098	0.068	-0.216
Speed-Straightness	-0.108	-0.169	-0.196	-0.447
Acceleration-TurningAngle	0.082	0.120	0.063	-0.099
Acceleration-Straightness	0.090	-0.033	0.055	0.008
TurningAngle-Straightness	-0.133	-0.354	-0.021	-0.083

2.5.3 Experiments

To demonstrate the utility of my proposed complexity features of movement in trajectory classification, two experiments are designed to evaluate the trajectory classification task. In the first experiment, all proposed movement features are used to build a classifier. In the second experiment, all but the complexity features are used to build another classifier. The same sample data from the data preprocessing stage are used to extract features, and to train and test the classification model in both experiments. The performance results of two experiments are compared and analyzed in the next section.

The goal of the experiments is to assign a correct movement type from four predefined categories to an unknown trajectory, which is a typical supervised multiclass classification problem. The "one-against-rest" strategy is used to build a multiclass classifier based on binary classifiers: four binary classifiers are built and each executes a binary classification of one movement type against the rest (e.g. walk vs. non-walk). They work together to compose a multi-class classifier by assigning the label of the highest prediction probability classifier to the unknown trajectory. For better classification performance, a 5-fold cross-validation approach is applied to evaluate all classifiers. As a result, in each run, 80% (320 trajectories) of preprocessed data are used to train this multi-class classifier, and 20% (80 trajectories) data are used for testing.

To select a suitable classification model in the experiments, different classification models, which include *C*-SVM (model parameters can be seen in next paragraph), *Nu*-SVM (kernel=radial basis function (RBF), nu=0.5, gamma=0.25, tolerance=0.0001, cost parameter=200), *k*-nearest neighbor (KNN) classification model (*k*=15 and Euclidean distance as weight), Logistic regression linear classification model (C=1e4, intercept scaling=2, penalty=12, tolerance=0.0001), Gaussian naïve Bayes model and Gaussian Mixture model (GMM) (alpha=0.1, number iterations=20, number components=4, threshold=0.01), are applied and compared to select the one with best performance. The results of cross-model comparisons of classification using dataset in first experiment are shown in table 2.4. According to the results, I employ the *C*-SVM, which achieves the highest prediction precision compared to four other classification models, to fulfill this classification task.

Table 2.4

Cross-model Comparisons of Classification Models Using Experiment 1 Dataset: 320 Training Data/80 Testing Data

	Precision	Recall	f1 score
C-SVM (RBF)	0.85	0.88	0.87
Nu-SVM (RBF)	0.83	0.82	0.82
KNN	0.71	0.71	0.71
Linear model (Logistic)	0.73	0.72	0.72
Naïve Bayes (Gaussian)	0.68	0.68	0.68
GMM	0.57	0.56	0.56

According to the form of the error function, there are two types of classification SVM: one is classification SVM Type 1 (*C*-SVM) and the other is classification SVM Type 2 (*Nu*-SVM) (Chang et al. 2001). Since *C*-SVM produces slightly higher prediction precision than *Nu*-SVM, I use the *C*-SVM empirically in both experiments. The parameters of this classifier are utilized automatically by sweeping all parameters within the valid range. Both experiments generate highest precision using the radial basis function (RBF) kernel. In the first experiment, the cost parameter is 128.0, complexity bound is 0.6, tolerance is 0.5 and numeric precision is 0.001. In the second experiment, the cost parameter is 32.0, complexity bound is 0.5, tolerance is 0.5 and numeric precision is 0.001.

2.5.4 Results

The results of the multi-class classification in the first experiment are shown in the confusion matrix in Table 2.5. As a result, the overall accuracy of prediction is about 85.4%, which is a good prediction result in multi-class trajectory classification outperforms much existing work (see review in section 2.2.2). Each entry in this matrix represents the proportion of true prediction. From the results, we can see that if the movement type of input trajectory is “car”, there is a 94.12% chance that the classifier assigns the correct label. The movement type “walk” also has high prediction accuracy (94.12%). The movement type “run” has the lowest prediction accuracy (72.92%). There is 18.75% chance to incorrectly predict it as “bike” and a 6.25% chance to incorrectly recognize it as “car”. The movement type “bike” also has a relative low accuracy (79.59%). Almost all movement types could be misclassified vis-a-vis the rest types, except that there is no misclassification of “car” to “run”. Such misclassifications require further investigation to improve the accuracy of the classification.

Table 2.5

Confusion Matrix of Accuracy for 4-class Trajectory Classification Problem in Experiment 1 (with Complexity Measures as Movement Features)

		Predicted Class			
		Bike	Car	Run	Walk
Actual Class	Bike	79.59%	10.20%	4.08%	6.12%
	Car	1.96%	94.12%	0.00%	3.92%
	Run	18.75%	6.25%	72.92%	2.08%
	Walk	1.96%	1.96%	1.96%	94.12%

In experiment 2, the overall prediction accuracy is 78.39%, which is lower than the overall accuracy of the classifier in experiment 1. This means that introducing the complexity measures of movement as features for trajectory classification can improve the overall prediction accuracy of the classification model. Specifically, the prediction accuracy has significant improvement in movement type “walk” (94.12% versus 88.24%), “run” (94.12% versus 80.39%) and “bike” (79.59% versus 61.22%) (see table 2.6). The ROC curves and area under the ROC curve (AUC) values of the two experiments also supports this conclusion (see table 2.7 and figure 2.4): the average AUC value of experiment 1 is higher than experiment 2 (0.917 versus 0.885), which demonstrates the good performance of the proposed classification model. The details of model comparison can be examined in the ROC curves and AUC values comparison of "one-against-rest" binary classification tests in figure 2.4. The comparison results show that the complexity measures of movement can be used to discriminate different types of movement and used as important features of movement to improve the accuracy of classification model.

It is also interesting to note that the prediction rate of "run" in experiment 2 is higher (83.33%) than in experiment 1 (72.92%). This means that incorporating complexity measures of movement has a negative effect on distinguishing "run" from other movement types. Further investigations are required to explain this underperformance. Considering that the differences are misclassified as "bike", it may be because complexity measures cannot discriminate "bike" and "run".

Table 2.6

Confusion Matrix of Accuracy for 4-class Trajectory Classification Problem in Experiment 2 (without Complexity Measures as Movement Features)

		Predicted Class			
		Bike	Car	Run	Walk
Actual Class	Bike	61.22%	4.08%	22.45%	12.24%
	Car	5.88%	80.39%	9.80%	3.92%
	Run	4.17%	6.25%	83.33%	6.25%
	Walk	1.96%	1.96%	7.84%	88.24%

Table 2.7

Comparison of Classification Results of Experiment 1 and 2

	Overall Accuracy	Overall Error Rate	AUC (average)
Experiment 1	85.42%	14.58%	0.917
Experiment 2	78.39%	21.61%	0.885

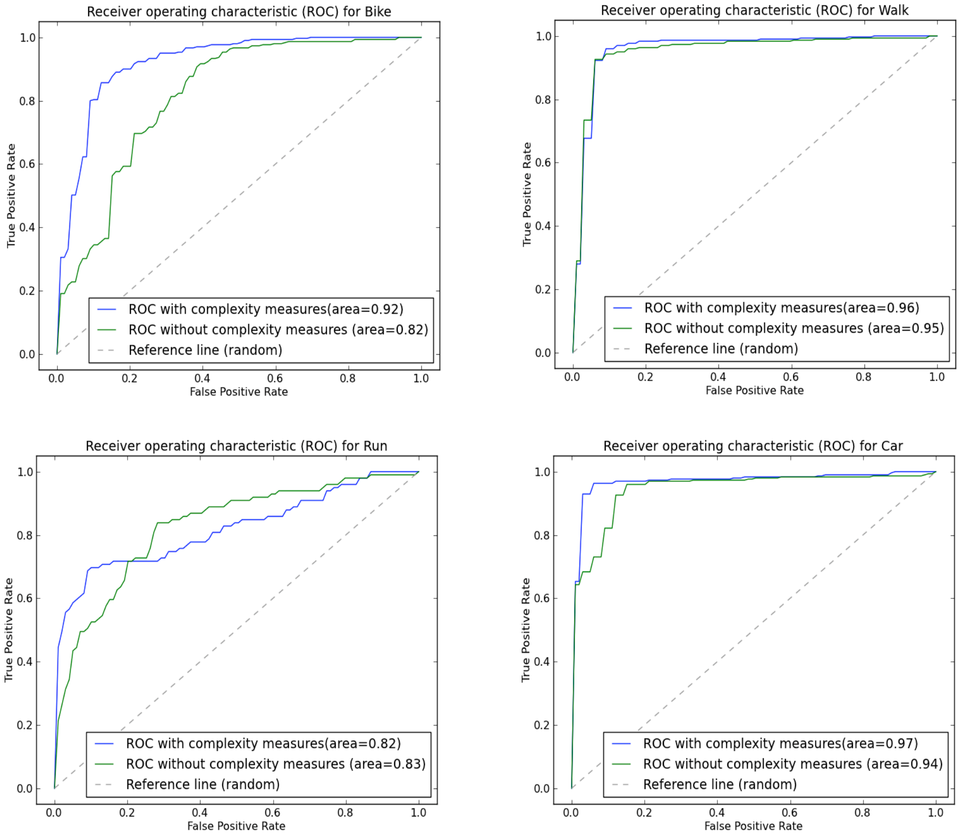


Figure 2.4: Receiver operating characteristic (ROC) and area under ROC curves comparison of "one-against-rest" binary classification results of Bike, Walk, Run and Car in experiment 1 and 2

2.6 Conclusion

In this essay, I presented a classification model based on effect movement parameters for automatically detecting movement types with unknown trajectories. To overcome some of the problems with the current approach based on movement parameters, I introduced the geometric complexity measures of trajectories and structural complexity measures of movement parameters as two new types of movement features for trajectory classification. These two types of complexity measures actually highlight both general geometric characteristics and the subtle changes of movement parameters that exist in different moving trajectories in a classification model. The results from two experiments demonstrate the positive effects of these complexity measure-based features in trajectory classification. Besides, this classification model overcomes two major problems in current research: (1) trajectory classification is limited to a certain spatial context when incorporating geometric shape characteristics in the classification model, and (2) trajectory classification can only be applied to the same size trajectories when incorporating local features from movement parameters.

Future research could focus on two additional aspects. First, the performance of classification related to different number of predefined classes and to large-scale data are important evaluating indicators for a multi-class classifier. The current essay only used four different movement types for trajectory classification. Other movement types, such as riding a motorcycle, should be included to assess the performance of this model. It would also be worth testing the performance of classification on large-scale data, since only 400 selected trajectories were used in this research. Second, the classification model proposed in this research is not sensitive to the length of trajectories. However, this is because the trajectory data that were collected for experiments were selected by filtering multi-behavior tagged trajectories. In practical applications, trajectory data could contain more than one behavior, which is beyond the scope of this classification model. Therefore, trajectory segmentation could be studied and used in data preprocessing to overcome this challenge.

2.7 References

Adrienko, N. & G. Adrienko (2010) Spatial generalisation and aggregation of massive movement data. IEEE Transactions on Visualization and Computer Graphics, 17, 205-219.

Ashbrook, D. & T. Starner (2003) Using GPS to learn significant locations and predict movement across multiple users. Personal and Ubiquitous Computing, 7, 275-286.

Bashir, F. I., A. A. Khokhar & D. Schonfeld (2007) Object trajectory-based activity classification and recognition using hidden Markov models. Image Processing, IEEE Transactions on, 16, 1912-1919.

Batty, M. (1985) Fractals-geometry between dimensions. New Scientist, 105, 31-5.

Bishop, C. 2006. Pattern recognition and machine learning. New York: Springer.

Buchin, K., M. Buchin, J. Gudmundsson, M. Löffler & J. Luo (20011) Detecting commuting patterns by clustering subtrajectories. International Journal of Computational Geometry and Applications 21, 253-282.

Chatfield, C. 2004. The analysis of time series: an introduction. Florida, USA: Chapman & Hall/CRC.

Dodge, S., P. Laube & R. Weibel (2012) Movement similarity assessment using symbolic representation of trajectories. International Journal of Geographic Information Science, 1-26.

Dodge, S., R. Weibel & E. Forootan (2009) Revealing the physics of movement: Comparing the similarity of movement characteristics of different types of moving objects. Computers, Environment and Urban Systems, 419-434.

Dodge, S., R. Weibel & A. K. Lautenschütz (2008) Towards a taxonomy of movement patterns. Information Visualization, 7, 240.

Eagle, N. & A. Pentland (2006) Reality mining: sensing complex social systems. Personal and Ubiquitous Computing, 10, 268.

Feldman, D. P. & J. Crutchfield. 1998. A Survey of "Complexity Measures". http://www.santafe.edu/~cmg/compmech/tutorials/ComplexityMeasures.pdf.

Fritz, H., S. Said & H. Weimerskirch (2003) Scale–dependent hierarchical adjustments of movement patterns in a long–range foraging seabird.

Proceedings of the Royal Society of London. Series B: Biological Sciences, 270, 1143.

Froehlich, J. & J. Krumm (2008) Route prediction from trip observations. Society of Automotive Engineers (SAE), 2193, 53.

Fu, Z., W. Hu & T. Tan. 2005. Similarity based vehicle trajectory clustering and anomaly detection. In IEEE International Conference on Image Processing, II-602-5. Genoa, Italy: IEEE.

Giannotti, F., M. Nanni, F. Pinelli & D. Pedreschi. 2007. Trajectory pattern mining. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 330-339. San Jose, California, USA: ACM.

Giannotti, F. & D. Pedreschi. 2007. Mobility, data mining and privacy: Geographic Knowledge Discovery. Berlin Heidelberg: Springer-Verlag.

González, M., C. Hidalgo & A. Barabási (2008) Understanding individual human mobility patterns. Nature, 453, 779-782.

Hanley, J. A. & B. J. McNeil (1983) A method of comparing the areas under receiver operating characteristic curves derived from the same cases. Radiology, 148, 839.

Hoffmann-Wellenhof, B., H. Lichtenegger & J. Collins. 2001. GPS: Theory and Practice. Wien: Springer.

Johansson, A., D. Helbing, H. Al-Abideen & S. Al-Bosta (2008) From crowd dynamics to crowd safety: A video-based analysis. Advances in Complex Systems, 11, 497–527.

Lee, J.-G., J. Han, X. Li & H. Gonzalez. 2008. TraClass: trajectory classification using hierarchical region-based and trajectory-based clustering. In Proceedings of the 34th International Conference on Very Large Data Bases, 1081-1094. Auckland, New Zealand: Endowment.

Lin, R. & H. Shim. 1995. Fast similarity search in the presence of noise, scaling and translation in time-series databases. In Proceeding of the 21th International Conference on Very Large Data Bases, 490--501. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Makris, D. & T. Ellis. 2002. Spatial and probabilistic modelling of pedestrian behaviour. In Proceeding of British Machine Vision Conference, 557-566. Cardiff, UK: BMVC.

Mamoulis, N., H. Cao, G. Kollios, M. Hadjieleftheriou, Y. Tao & D. Cheung. 2004. Mining, indexing, and querying historical spatiotemporal data. In Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 236-245. New York, NY, USA: ACM.

Mandelbrot, B. (1967) How long is the coast of Britain? Statistical self-similarity and fractional dimension. Science, 156, 636.

Nams, V. (1996) The VFractal: a new estimator for fractal dimension of animal movement paths. Landscape Ecology, 11, 289-297.

Nams, V. O. (2005) Using animal movement paths to measure response to spatial scale. Oecologia, 143, 179-188.

Nara, A. & P. M. Torrens. 2007. Spatial and temporal analysis of pedestrian egress behavior and efficiency. In Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information System, 1. New York, NY, USA: ACM

Nguyen, N. T., D. Q. Phung, S. Venkatesh & H. Bui. 2005. Learning and detecting activities from movement trajectories using the hierarchical hidden Markov model. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 955-960. San Diego, California, USA: IEEE.

Niu, W., J. Long, D. Han & Y. F. Wang. 2004. Human activity detection and recognition for video surveillance. In Proceedings of the 2004 IEEE International Conference on Multimedia and Expo, 719-722. Taipei, Taiwan: IEEE.

Pincus, S. (2008) Approximate entropy as an irregularity measure for financial data. Econometric Reviews, 27, 4, 329-362.

Pincus, S. M. (1991) Approximate entropy as a measure of system complexity. Proceedings of the National Academy of Sciences, 88, 2297-2301.

Shannon, C. E. (1948) A mathematical theory of communications, I and II. Bell System Technical Journal, 27, 379-423 and 623-656.

Smith, L. I. 2002. A tutorial on principal components analysis. www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf.

Torrens, P. (2008) Wi-fi geographies. Annals of the Association of American Geographers, 98, 59-84.

Torrens, P., X. Li & W. A. Griffin (2011) Building agent-based walking models by machine-learning on diverse databases of space-time trajectory samples. Transactions in GIS, 15, 67-94.

Vlachos, M., M. Hadjieleftheriou, D. Gunopulos & E. Keogh. 2003. Indexing multi-dimensional time-series with support for multiple distance measures. In Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 216. Washington, DC, USA: ACM.

Yazji, S., R. P. Dick, P. Scheuermann & G. Trajcevski. 2011. Protecting private data on mobile systems based on spatio-temporal analysis. In Proceedings of the International Conference on Pervasive and Embdeded Computing and Communication Systems, 114-123. Vilamoura, Algarve, Portugal: PECCS.

Zheng, Y., L. Zhang, X. Xie & W. Ma. 2009. Mining interesting locations and travel sequences from GPS trajectories. In Proceedings of the 18th International Conference on World Wide Web, 791-800. New York, NY, USA: ACM.

Essay 3

Using ESDA with Social Weights to Analyze Spatial and Social Patterns of Preschool Children's Behavior

Abstract

To study the development of social behavior of preschool children, micro-social data of preschool children were collected for the first time in both a space and time context using a novel behavioral coding system. These unique space-time micro-level behavioral data enable us to explore the group-level, dynamic, social, and socio-spatial patterns of children's behavior from both a geographic and a social perspective. In this essay, GIS, spatial analysis, and social network analysis techniques are for the first time employed together to study group-level social behavior emerging from children's everyday activities and interactions. A methodology called exploratory spatial data analysis with social weights, which can be applied in both geographic and social space, is applied to explore dynamic social and socio-spatial patterns of preschool children's behavior. This methodology is used to analyze and visualize group-level dynamic spatial and social patterns of preschool children's behavior based on long-term spatiotemporal behavioral observations. The spatial and social analysis generates several interesting results of dynamic social and spatial patterns of preschool children, which highlights the utility of this approach for analyzing social dynamics among preschool children. This research also provides social scientists with a powerful software toolkit to study the relationship of preschool's spatial settings and activities in regards to the socialization of preschool children.

3.1 Introduction

Childhood is recognized as a key stage in the development of human social behaviors (Holt 2007). Studying the socialization of children is important for investigating how human social behaviors develop and for understanding the evolution of complex social phenomena. Children's social skills and behaviors are mostly developed by playing with peers, either spontaneously or guided by parents or teachers (Rogers and Sawyers 1988). Children's play is impacted by many factors, including children's characteristics (Fishbein and Imai 1993, Leman and Lam 2008), activity settings (Oden and Asher 1977) and environmental settings of play (Barbour 1999). In this context, questions of who children play with, how they play, and where they play are relevant. To address these questions, children's characteristics, peer-to-peer interactions, activities and social behaviors can be recorded with a space and time dimension. To my knowledge such spatiotemporal behavioral data for children has not been collected before. Redesigning existing behavioral coding systems to record such data could provide new opportunities to explore, analyze, and visualize the group-level spatial and social patterns of preschool children's behavior from a joint geographic and social perspective.

Group-level patterns of human behavior are key for understanding human social behavior. They provide insights into how aggregate social outcomes are generated. These group-level patterns are also important for validating social dynamics in cases where it is questionable to model results at a macro-level perspective. Many studies have focused on finding patterns of social behavior using analytical or statistical methods (Sayer 1992). However, existing studies on preschool behavior are usually restricted to either a social or spatial context. For instance, geographic data and methods are usually used for discovering spatial patterns (spatial heterogeneity, spatial externalities and spatial spillovers) while social data and methods are mostly used for discovering social network patterns in preschool behavior. In this essay, I overcome this gap between methods in this literature by jointly applying both spatial and social methods to examine the relationship between geographic and social settings and children's social structure.

To overcome the limitation of separate spatial and social approaches and leverage the strengths of both approaches within the same framework, I build on recent research in other fields that combines geographic and social network analysis methods (Parker and Asencio 2008, Radil et al. 2010, Tita and Radil 2011). By doing so, spatial autocorrelation analysis, which is powerful in statistical analysis and visualization of spatial patterns, can be applied to analyze and visualize patterns of preschoolers' social behavior in new ways. Research questions in social and spatial contexts can be answered at the same time in this relatively new exploratory manner by providing social scientists a set of useful statistical metrics and pattern discovery methodologies for examining aggregate level social and spatial patterns of children's social behavior. The gap between social data analysis in the social sciences and spatial data analysis in the geographic sciences is bridged through an application of this hybrid methodology that is emerging in fields such as criminology or education but has not yet been applied in research on children's behavior. This essay demonstrates that this hybrid spatial and social methodology allows for an improved study of group-level socialization patterns that emerge from children's social behavior.

This essay adds to this new literature on joint spatial and social analysis by proposing a novel scheme for studying social behavior of preschool children. In this scheme, a behavioral coding system integrated with space-time GIS is designed to record longitudinal micro-social data, including behavior in time and space of preschool children. To analyze these spatiotemporal behavioral data as part of an exploratory approach to better understand group-level social phenomena in preschool children, I apply so-called exploratory spatial data analysis (ESDA) with social weights. ESDA with social weights integrates geographic space and social space of agents in a joint framework, provides spatial autocorrelation pattern analysis in both a spatial and social context, and can be used to analyze, discover and visualize spatial and social patterns from micro-social data. Specifically, I apply ESDA with social weights to explore preschool children’s social and spatial patterns to simultaneously examine the influence of geographic factors on children’s social behaviors and non-geographic factors on children’s social structure. I then compare these patterns for male and female preschoolers.

This essay is organized as follows: a review of related research of the development of social behavior of preschool children is provided first. Then I describe the participants, the study area, the design of the space-time behavioral coding system and the micro-social data we collected. Next, I present ESDA with social weights and how it is applied to explore socio-spatial patterns and social patterns of preschool children's behavior. In the subsequent section, I apply ESDA with social weights to real data and analyze the statistical and visualization results. The framework for analyzing the relationship between physical environment and activity settings in regards to preschool children’s social behavior helps to better understand the development of social behavior of children. I conclude this essay with a discussion of the findings, possible application extensions, and future work. The methodology and software presented in this essay is not restricted to the analysis of preschool children's behavior but is also applicable to other spatiotemporal micro-social data.

3.2 Related Work

Examining the impact of the development of social skills and behaviors of young children has been a long-standing area of research in the social sciences. In the last decades, studies have focused on examining the development of children’s social behavior from biological, socio-cultural, behavioral and geographical perspectives. For example, much research examined the effect of children’s characteristics, such as age, and sexuality on their social behavior (see Whiting and Edwards 1992, Rose and Rudolph 2006, Turner 1991). Other research studied the influence of socio-cultural characteristics, such as race, culture, social exclusion (disabilities), on the formation of socialization in children (Whiting and Edwards 1992, Gresham et al. 2001, Shores et al. 1993). Further, authors demonstrated that activity settings, such as teacher-oriented tasks, collaborative tasks and individual tasks of playing (e.g., playing with the computer), have significant effects on children's development of (normal/abnormal) social skills and behaviors (see Quilitch and Risley 1973, Oden and Asher 1977, Rogers and Sawyers 1988, Plowman and Stephen 2005). Spatial structure is also significant in framing sociality because it can capture the clustering of individuals, dyads, and groups, as well as their configurations, such as proximity, spatial cohesion, peripherality etc., in space (Griffin et al. 2007). Research has examined the significant influence of the environmental settings of play materials on children's social behavior (see Barbour 1999, Gutierrez Jr et al. 2007).

The methods in such observational research of children's social behavior usually collect data from longitudinal individual-level behavioral observations, and then utilize classic qualitative or quantitative techniques to study the patterns of social behavior in children. For example, based on observation of free playing, Strayer and Stantos (1996) applied network analysis methods to analyze the social structure to assessing the degree of social stratification of preschool children. In, Vaughn BE et al. (2001), the analysis of variance (ANOVA) statistical models are employed to test the friendship relationship using dyadic observations in a sample of preschool children. In Stantos et al. (2008), hierarchical cluster analysis is applied for identifying affiliative subgroups in preschool children, and studying the social structures. In Griffin et al. (2007), geographic analysis techniques are applied to study the micro-social patterns of preschool children. However, to my knowledge, relatively little work has been done in studying the dynamics of socialization of children, despite the theories and methodologies of studying social dynamics of adults are increasingly applied in areas such as public health, public safety, urbanization and transportation (Epstein 2006). What has been overlooked is the mechanisms of how individual social behaviors and dyad interactions of children translate into social phenomena, such as the social structures or spatial structures underlying the everyday practice of children. Examining the macro patterns emerging from social behaviors at micro scale is important for understanding such social mechanisms by improving knowledge of socialization from macro spatial and social perspectives.

In regional science, a spatial perspective has been explicitly considered for studying social behavior of individual agents in various forms, such as peer effects, neighborhood and network effects (Anselin et al. 2004). In these studies, spatial correlation is incorporated into spatial models to better deal with individual social behavior. For example, research on the clustering patterns of human behavior demonstrates that social interactions are not only related to socio-economic distance but also related to geographic distance between agents (Conley and Topa 2002). Other research shows that spatial differentiation of neighborhoods plays an important role in spatial patterns (e.g. clustering, diffusion, contagion) of social behaviors such as crime (Messner and Anselin 2004), school performance (Fotheringham et al. 2001) or migration (Boots and Kanaroglou 1988).

Recently, spatial statistical analysis has been applied to study human social behavior in the context of geographic space and its spatial patterns at aggregate scales in the social sciences such as criminology, political science, public health, and economics (see a review in Anselin 2010). By mapping social behavior and other related information in physical space, spatial statistical techniques can be used to examine the geographic characteristics of social behavior in space and to study the impact of the geographic context on the outcomes of the human behavioral patterns of interest. Patterns that are related to processes of diffusion and contagion of human social behavior or that result in externalities can be identified and tested using spatial statistical analysis.

To efficiently explore social behavior data in a geographic context, exploratory spatial data analysis (ESDA), a subset of exploratory data analysis, has been developed as an important technique to describe and visualize spatial distributions, discover spatial patterns and suggest spatial regimes by using spatial statistical analytical approaches (Anselin 1994). Key to ESDA is the notion of spatial autocorrelation or spatial association: the phenomenon where locational similarity (observations in spatial proximity) is matched by value similarity (attribute correlation). It can be utilized to measure the overall spatial clustering and to detect local clusters and outliers (Anselin et al. 2007).

Spatial effects and spatial association have also recently been extended and successfully applied to study network correlated behavior in non-geographic space--this is aided by similarities between the conceptualization and measurement of spatial correlation and network correlation (Leenders 2002). For example, Black (1992) discusses extending Moran’s I to assess the existence of network autocorrelation in transport network and flow system network space; Yamada and Thill (2007) extend the local Moran’s I statistic to observations on a network as part of network-based spatial analysis and local-scale spatial data analysis on road networks. In social network analysis, Marsden and Friedkin

(1993) argue that social influence between agents in a social network should be considered a “network effect” in modeling autocorrelated social processes. These “network effects” are similar to “spatial effects”. Leenders (2002) further discusses a series of operationalizations of the weight matrix W and parameter estimates and inferences of such social autocorrelation models. Further, Farber, Páez et al. (2009) investigate the ability of spatial dependency tests to identify a spatial or network autoregressive model. The increasing number of new applications of spatial autocorrelation and network autocorrelation in the social sciences (Anselin 2010, Tita and Radil 2011) indicates the potential of using this joint spatial-social framework for an improved understanding of preschool children’s social behavior.

However, there is a research gap in jointly considering spatial and social patterns in a space-time context. Many existing data analysis methods in spatial analysis and social network analysis have been developed to discover patterns related to specific social behaviors such as crime, epidemic, or Internet-based social media. Such research tends to focus on either spatial or social patterns and either space or time. Although joint space-time, spatial-social pattern detection is seen as useful and important for better understanding socialization and social dynamics, these dimensions are nevertheless often viewed in isolation (Radil et al. 2010). These two aspects of data analysis have rarely been employed together for studying aggregate-level human behavior from a geographic and social perspective, and have not been used to study preschool children’s social behavior to the author's knowledge.

3.3 Methods

ESDA with social weights as applied in this essay utilizes a collection of graphical and statistical techniques to visualize social and spatial distributions, and to discover group-level social and spatial patterns of preschool children’s behavior over time. By focusing on both social and spatial aspects of the social behavioral data, this hybrid methodology has several benefits for better understanding the socialization development of preschool children. First, it provides a toolset with functionalities to analyze potential patterns statistically and visualize them graphically for ease of interpretation. Second, mature techniques in spatial analysis and geovisualization are used to focus on studying the spatial dimensions of children's social behavior. Third, the relatively recent extension of spatial association to network association provides a powerful statistical and visualization approach to examine the non-geographic factors of socialization. Finally, a combination of social and spatial analysis makes it possible to answer the questions of social behavior development in an integrated social and spatial context.

In ESDA with social weights, spatial autocorrelation extended to social space using social distance weights is the core technique to explore spatial and social cluster patterns at a global and local level. Cluster detection is one of the most common aggregate-level patterns in both spatial and social space, and can be used to examine important aspects of social behavior such as identifying outliers, finding group structures, determining similarities and discovering hotspots. Further, ESDA with social weights as applied in this essay also provides tools for traditional exploratory data analysis plus a social and spatial density surface analysis and space-time visualization for all patterns.

3.3.1 Spatial Autocorrelation and Local Indicators of Spatial Association

When spatial autocorrelation is present, observations with similar values are also proximate in space. Global spatial autocorrelation is a measure of overall spatial clustering in the observations. It is evaluated by testing against a null hypothesis of a random spatial pattern of observations. Rejection of this null hypothesis suggests spatial structure, which in this context means a propensity for social activities to correlate with other social activities in space. This is significant for examining the effect of environmental settings on social behavior, as social interactions are often catalyzed by affinities in behavior in the same place.

Moran's I is a classic measure for spatial autocorrelation. The global version of this statistic can be used to diagnose the presence of overall clustering in a study area. The global Moran index is calculated as follows (Anselin 1996):

$$I = \sum_i \sum_j w_{ij} \cdot (x_i - \mu) \cdot (x_j - \mu) / \sum_i (x_i - \mu)^2 \tag{3.1}$$

where w_{ij} is the row-standardized contiguity matrix, x_i is the frequency count of a certain behavior at location i , and μ is the average frequency count of this behavior. Its statistical significance can be evaluated based on a comparison to a reference distribution obtained by randomly permuting the observed values in space several times. The values of Moran's I are not constrained by –1 and +1 since they depend on the weights matrix. A zero value indicates that no spatial autocorrelation is present. In this study, it means that a given social activity would tend to be randomly distributed over space, and has no clustering effects. Negative/positive values indicate negative/positive spatial autocorrelation. Negative spatial autocorrelation, which reflects a checkerboard pattern, means that social activities tend to repel themselves in space. Positive spatial autocorrelation means that social activities tend to cluster in space.

The spatial weights w_{ij} are a central component of the Moran's I test. They define which locations are neighbors, i.e. spatially connected. Three popular spatial weights include contiguity-based weights, distance based weights and kernel weights.

To determine the location of clusters or spatial outliers, local indicators of spatial association (LISA) are employed by applying a local Moran statistic for evaluating spatial autocorrelation at the level of every observation. A local version of the Moran's I statistic is calculated as follows (Anselin 1995):

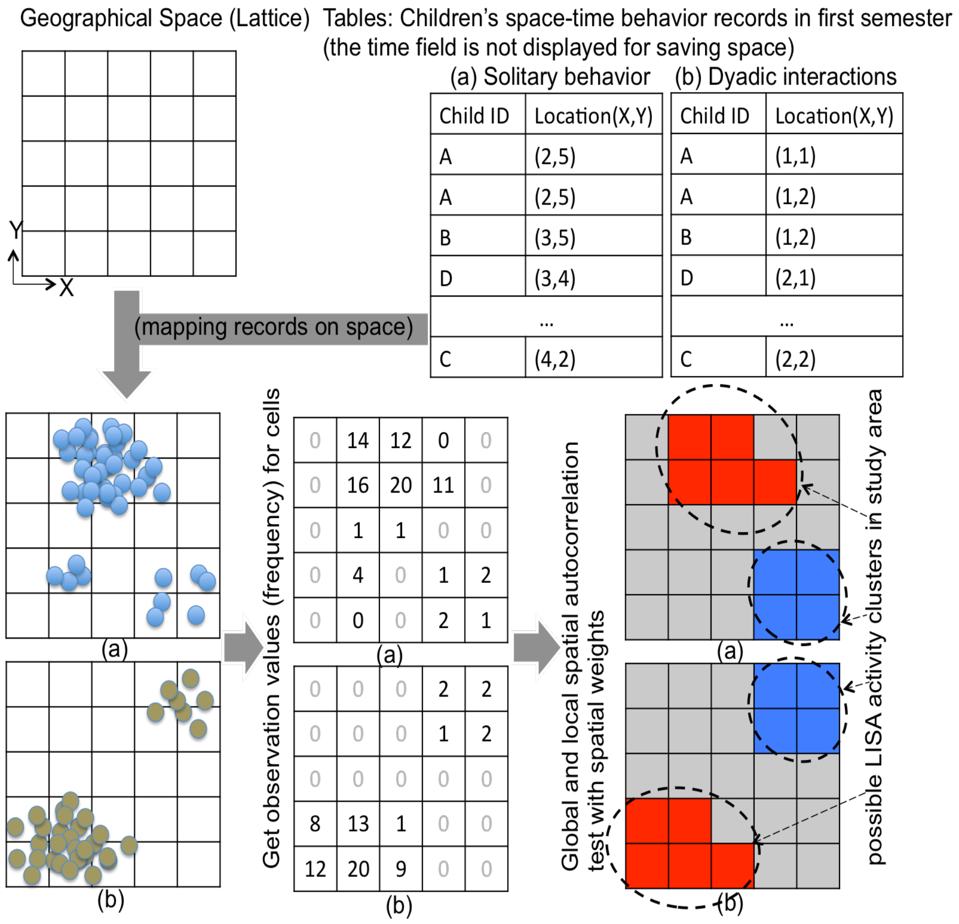
$$I_i = \frac{(x_i - \mu)}{\sum_i (x_i - \mu)^2} \sum_j w_{ij} \cdot (x_j - \mu) \qquad (3.2)$$

The local Moran allows us to examine the presence of local spatial patterns by classifying the local Moran statistic into four groups: high-high and low-low (values were found to be surrounded by similar neighboring values above or below the mean) which represent local spatial clusters, and high-low and low-high (values were found to be surrounded by dissimilar neighbors) which represent local spatial outliers. A map highlighting the significant local spatial clusters at geographic locations is called a LISA cluster map, which is useful for identifying interesting locations (geographical settings) and to evaluate spatial heterogeneity of attributes (social behavior). In this research, high-high and low-low clusters are particularly useful for studying children’s social behavior. High-high values are also called “hotspots” to reflect that certain behaviors frequently occur in these cluster areas. In a social setting, “hotspots” indicate spatially correlated social behaviors. Low-low values are called “coldspots”, which means lower-than-average social activity levels can be found in a cluster area, possibly implying that the spatial settings in this area might have an inhibiting impact on social behavior.

In this essay, I first apply spatial autocorrelation tests with traditional geographic weights to study children's engagement in four different behaviors, which are solitary behavior, teacher oriented interaction, social interaction and parallel interaction, in indoor classroom and outdoor playground settings for two selected semesters. Specifically, to test traditional global and local spatial autocorrelation, the geographical study area (preschool in this research) is first divided into a lattice structure (see “geographical space” in figure 3.1). Then, for each behavior and each semester, children’s space-time records will be mapped to cells in this lattice space (see table (a) and (b) in figure 3.1). The number of records in each cell is the input value for the subsequent spatial analysis of these data. Each behavior type is characterized by a unique spatial distribution in this lattice space. Last, global and local spatial autocorrelation tests with spatial weights are used to analyze possible global clustering and local LISA clusters of each behavior for each semester in the study area (see graph (a) and (b) in figure 3.1).

3.3.2 Global and Local Spatial Autocorrelation with Social Weights

Further, I apply global and local spatial autocorrelation tests with social network weights to determine if the frequency of engaging in an activity is autocorrelated. However, in contrast to the case illustrated in figure 3.1, here the autocorrelation test is not applied in geographic space but in social network space, i.e. based on children's frequency of playing with each other. Hence, in contrast to the typical geographic applications of spatial autocorrelation tests, I define spatial relations with social distance weights. And, diverging from the typical social network analysis, I do not consider whether children engage in an activity with each other but ask whether children who play a lot with each other engage in similar activities (but not necessarily with each other). In more technical terms, my application of spatial autocorrelation with social weights tests whether the frequency with which children engage in an activity (independently of who they play with) is correlated with the frequency with which children play with each other. This test can be used to answer the research question: is what children's intra-person propensities associated with their inter-person social relationship?


Geographical Space (Lattice)
Y
X
(mapping records on space)
(a)
(b)
Get observation values (frequency) for cells
(a)
(b)
Global and local spatial autocorrelation test with spatial weights
(a)
(b)
possible LISA activity clusters in study area

Tables: Children's space-time behavior records in first semester (the time field is not displayed for saving space)

(a) Solitary behavior

Child ID	Location(X,Y)
A	(2,5)
A	(2,5)
B	(3,5)
D	(3,4)
…	
C	(4,2)

(b) Dyadic interactions

Child ID	Location(X,Y)
A	(1,1)
A	(1,2)
B	(1,2)
D	(2,1)
…	
C	(2,2)

(a)

0	14	12	0	0
0	16	20	11	0
0	1	1	0	0
0	4	0	1	2
0	0	0	2	1

(b)

0	0	0	2	2
0	0	0	1	2
0	0	0	0	0
8	13	1	0	0
12	20	9	0	0

Figure 3.1: Process diagram to illustrate global and local spatial autocorrelation with spatial weights as applied in this essay.

Specifically, to test global and local spatial autocorrelation with social weights, a spring-layout graph structure in social space is first created to represent the social network (see "spring-layout graph" in figure 3.2). It is based on the frequency data of social interactions between children (see table (i) in figure 3.2). This social network structure will be used as a "base map" in social space. Observations in this "base map" are node values that represent the frequency counts of children engaged in a specific activity (see table (ii) in figure 3.2). For each activity, a unique set of node values is presented for the same social network structure. Last, global and local spatial autocorrelation tests with social weights are used to analyze possible global clustering and local LISA clusters to examine the correlation between activity frequency counts and the underlying social network structure (see graph (a) and (b) in figure 3.2).

3.3.2.1 Social Distance and Social Networks

To extend the concept of spatial autocorrelation to the social network domain, some concepts in a social context should be clarified. In a social context, social agents are the units of analysis. They are socially connected with others to form a specific social structure, or in another words, a social network. The distance between two agents in a social network is called social distance, which represents how socially close they are. There are many ways to calculate social distance (Bogardus 1925, Brewer et al. 1987, Leenders 2002). In this research, an inverse normalized frequency count of social interactions between observations is used to calculate social distance. This is based on an intuitive assumption that more interactions happened between children, the closer they are with shorter social distance. For example, for a child A, whom played with B 50 times, D 40 times, and E 10 times, the normalized data, which are 50%, 40%, and 10%, represents the relative social relationship of A with his/her friends B, C and D. The social distances from A to his/her friends are simply calculates as the inverse values: $(50\%)^{-1}=2$, $(40\%)^{-1}=2.5$, and $(10\%)^{-1}=10$.

Then, this social network can be represented in form of a graph structure (see figure 3.2) where N is a set of nodes (social agents) and E is a set of edges where each edge: edge(i, j) connects two nodes (node$_i$ and node$_j$). For visualization purpose, the length of the edge is defined by the average social distance between agent *i* and *j:* {edge(*i, j*), edge(*j, i*)}. This base graph represents interactions between children where children with the most cumulative interactions per time period are close to each other in the center of the graph and children with the fewest interactions are on the periphery. If children interacted with each other during the given time period, this interaction is marked by a line (e.g. in figure 3.2, A had 3 lines, i.e. interacted with B, D and E, while G has 1 line, i.e. only interacted with C). No lines between children means there were no interaction. There is a separate base graph for each semester since the number of children and their interactions varies between semesters. At the same time, the number of interactions between children impacts the position of them in the graph: more interaction means children are closer but there is only one line to connect two children if more than one interaction occurred.

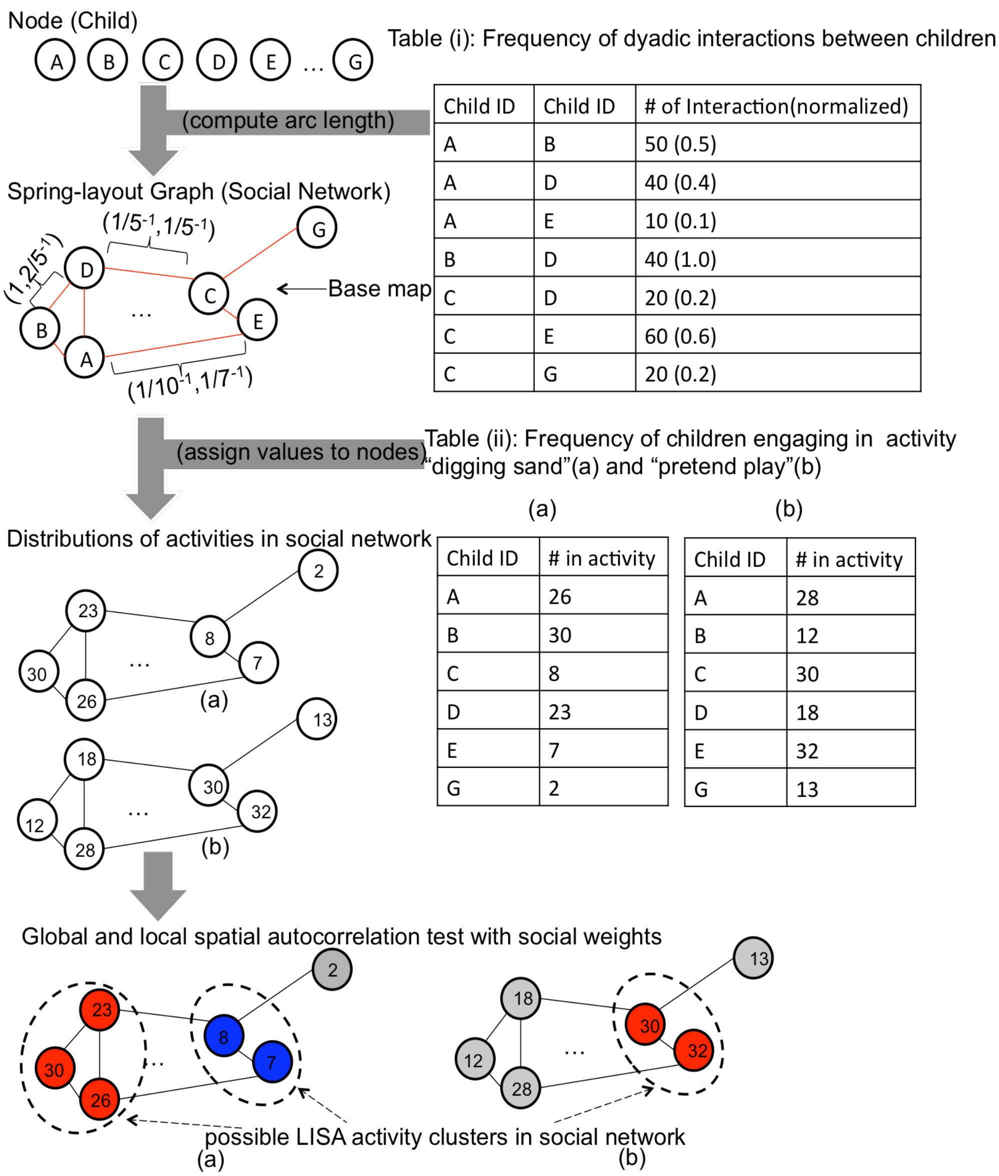

Node (Child)
A B C D E … G
(compute arc length)
Spring-layout Graph (Social Network)
(1/5-1,1/5-1)
(1,2/5-1)
G
D
C
B
A
E
…
Base map
(1/10-1,1/7-1)
(assign values to nodes)
Distributions of activities in social network
2
23
8
30
7
26
(a)
13
18
30
12
32
28
(b)
Global and local spatial autocorrelation test with social weights
2
23
8
30
7
26
13
18
30
32
12
28
possible LISA activity clusters in social network
(a)
(b)

Table (i): Frequency of dyadic interactions between children

Child ID	Child ID	# of Interaction(normalized)
A	B	50 (0.5)
A	D	40 (0.4)
A	E	10 (0.1)
B	D	40 (1.0)
C	D	20 (0.2)
C	E	60 (0.6)
C	G	20 (0.2)

Table (ii): Frequency of children engaging in activity “digging sand”(a) and “pretend play”(b)

(a)

Child ID	# in activity
A	26
B	30
C	8
D	23
E	7
G	2

(b)

Child ID	# in activity
A	28
B	12
C	30
D	18
E	32
G	13

Figure 3.2: Process diagram to illustrate the application of global and local spatial autocorrelation with social weights in this essay.

To visualize this social network, I use a “spring layout” force-based graph drawing technique, where the position of nodes are impacted by the weight (length) of an edge and a random initial direction. Springs connect these nodes, place them in a location and then adjust the positions of nodes until their relative positions reach a balance on the spring force, which connects agents in the following way: socially close agents are also closely related to each other in the graph. This graph layout represents the basic social structure of social agents, in this case, preschool children. I apply a “spring layout” algorithm to visualize the children’s social network since it is particularly popular in representing social networks. However, how to visualize the social network doesn’t impacts the underlying social structure and the weights matrix (the latter is computed based on the social structure). Therefore, the “spring layout” algorithm may generate different network layouts because of using different initial direction that impacts the angle from which the spring layout is viewed, but the relative positions of nodes remain the same in the social structure. In this sense, the layout of the social network is relatively "robust" for social distance weights since these weights only take the relative position into account.

3.3.2.2 Spatial Autocorrelation with Social Weights

In this application of spatial autocorrelation (Moran's I) with social weights, I test whether activity engagement is correlated among children who are socially close but do not necessarily engage in an activity together. A significant Moran's I value suggests similarity in activity engagement for children who play a lot with each other, i.e. social proximity of agents is matched by their propensity similarity of activities. This is relevant for examining the relationship between agents' activity preference and social structure because children’s intra-person propensity of activities could be associated with the formation of inter-person social structures. If there is a significant positive spatial autocorrelation for a specific activity among children who are socially close, then this type of activity might also cluster in social network space. In other words, it suggests the hypothesis, which could be tested in further research, that this activity plays a positive role in regards to socialization. Therefore, spatial autocorrelation with social weights can be a useful statistical metric to inform subsequent tests of which activities can lead to more social interactions (form social structures) and are conducive for developing social skills.

To compute the Moran’s I statistic for spatial autocorrelation with social weights, social weights need to be defined since the social context is based on social rather than geographic proximity. For the social weights, the base graph of a social network is used similarly to a base map of geographically defined nodes except that connectivity is defined with a social rather than geographic weights matrix. Each node is associated with the frequency that child *i* engaged in an activity (15 fields of activities) in a certain time period. The relative position of each node is determined by connectivity that is measured as inverse normalized frequency counts of social interactions with other children. This measure is the weight value from one node to another node. Connections in this social weights matrix indicate that children interact with each other.

It is worth noting that popular methods for deriving spatial weights, such as contiguity-based weights, cannot be applied directly in our social network because every child usually interacted (is contiguous) with every other child. Two different social distance-based weights for comparison purposes are considered in this research. The first are *k*-nearest neighbors (KNN) weights. In this approach, for each child, I sort her/his friends by interaction frequencies, then pick the first *k* friends as neighbors. Other common options include distance bands and inverse distance. I hence also apply inverse distance weights which assume that every child plays with every other child but less so with more distant children (where distance is defined as having fewer interactions between children). Since distance bands would be based on a threshold of how many friends a child interacts with in social space, this metric is similar to the *k* nearest neighbor approach used above and hence not included. Therefore, I adopted the *k*-nearest neighbors (KNN) weights in the experiments. After defining social weights, the global Moran's I statistic can be calculated to examine if the frequency with which children engage in a particular activity (regardless of who they play with) is correlated with the frequency with which children play with each other.

3.3.2.3 LISA with Social Weights

In the original definition of LISA, space can be defined geographically, socially or in terms of other metrics (Anselin 1994). In this essay, LISAs are applied in a social context by using social weights as defined above to examine activity clusters for children who are socially close. Again, this differs from traditional social network analysis that tries to study which children engage in a particular activity with each other. The local Moran statistic, an example of a LISA, is used for locally evaluating the significance of activity autocorrelation for every social agent. The local Moran's I is calculated in the same way as in the geographic case but using social instead of spatial weights (see section above). LISAs with social weights can detect the location and the type of activity cluster patterns in a social network. More specifically, in the case of this study of children's behavior, LISAs with social weights can be used to indicate what activity is associated with which part of the social network and where activity "hotspots", "coldspots" and outliers are located. For example, for child *i* with *k* neighboring children, one determines if child *i*'s activity engagement values are correlated with the average activity values of its closest *k* neighbors (i.e., the *k* children s/he interacts with most but not necessarily in this activity) beyond what one would expect under conditions of social randomness. If this is the case, the LISA map core is displayed as significant.

The LISA map cores are classified into four categories that represent four different activity patterns. In this research, these activity clusters for socially connected children are used for studying children's social behavior. A network graph that highlights social agents with significant local Moran indices and corresponding activity cluster patterns in a spring layout formis referred to as a LISA cluster graph here. It can be used to identify interesting social groups (agents in social proximity) and to assess the extent to which activity behavior exhibits heterogeneity in a social network graph (with clusters of high interactions and low interactions).

3.4 Data, Experiments and Results

3.4.1 Data

For collecting spatiotemporal micro-social data of preschool children in a single urban American preschool (see figure 3.3), a TabletPC based behavioral coding systemis originally developed by Griffin et al. (2007). This system allows coders to use a digital pen to record preschool children's behavior in a GIS-based graphic user interface (see figure 3.4). The data collection took place over a five-semester period (from Fall 2007 to Spring 2009) generating 184,000 observations of interactions that traced the dynamic development of sociality in 84 preschool children[8]. For the experiment in this research, I select a subset of data from last two semesters (Fall 2008 and Spring 2009), which I think has relatively high quality because of the maturity of coding procedures. This experimental data contains 34,657 records observed from 38 preschool children.

Children's behavior data were collected for five-and-a-half hours each day. Approximate three observers worked simultaneously. Observers identified children in a randomized list and observed a child for ten seconds and then recorded data. This procedure is then repeated for the next randomly selected child. Specifically, coders recorded the time and geographical location of the child, whether the child was alone (solitary behavior), with a teacher (teacher

[8] This work was funded by the National Science Foundation as part of the project "Modeling time, space, and behavior: Combining ABM & GIS to create typologies of playgroup dynamics in preschool children" by William Griffin, Paul Torrens, Jennifer Fewell (2006-2011). The data were collected by Casey Sechler, Jillian Smith and other ASU undergraduate and graduate students who have participated in this project.

interaction), directly engaged in a group (social interaction), interacted with other children (peer interaction), passively or loosely engaged in group behavior through parallel play (parallel interaction, where children are playing in proximity to each other, but not with each other) and the activity the child was engaged in.

For four different behaviors, the target child was observed for one of 15 activities (see table 3.1). In this study, the data were divided into two parts for two semesters. In each semester, the proposed method will be applied to analyze spatial and social patterns of children’s behavior. For an overview of these data, figure 3.5 presents a separate kernel density map of all observed children's behaviors for the Fall 2008 and Spring 2009 semester. In each density map, the output cell size is defined as 0.1 feet and the search bandwidth is 3.0 feet.

Table 3.1

15 Observed Activities with Brief Definitions

	Task Name	Description
1	Art	coloring, painting, collage, gluing
2	Board Games	candyland, ants in the pants, connect four, playing cards
3	Digging	digging sandbox, garden
4	Figure Play	dolls, action figures, people figures, toy animals
5	Language Arts	books, writing, books on tape
6	Large Motor	running, climbing, swinging, bikes, wagon
7	Manipulatives	blocks, legos, lincoln logs, connects, puzzles
8	Math/Science	magnets, counting bears, space theme, balance scale
9	Molding	play-dough, goop, clay
10	Music/Singing	listening to the radio, singing, dancing to music, playing instruments
11	Physical Games	ring around the rosey, red rover, tag, sports
12	Pretend Play	"getting married", "being Superman", playing kitchen, dress up with a theme
13	Sensory Play	shaving cream, water, bubbles, dump and pour materials like corn kernels
14	Talk	conversation – if they are talking about what they're doing
15	Walking	moving between locations – do not have to know the destination, just distinguish from aimless walking

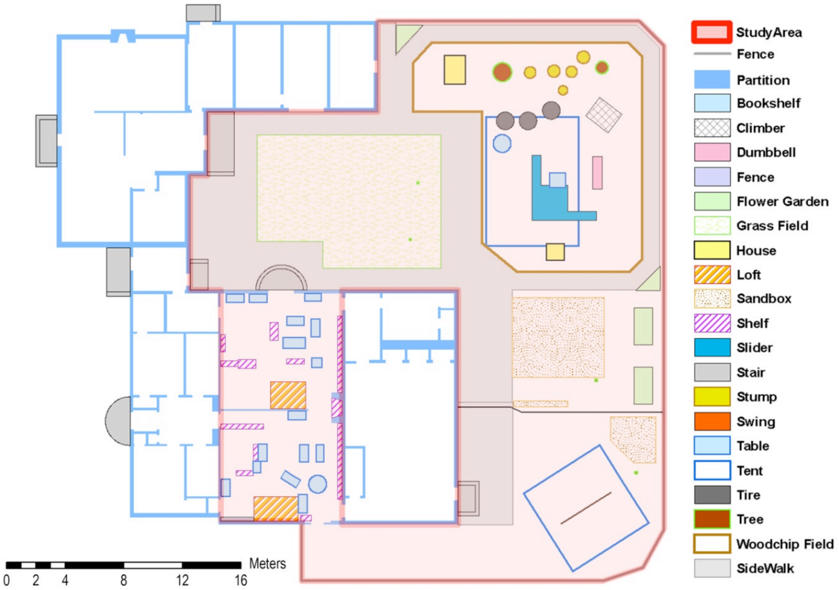
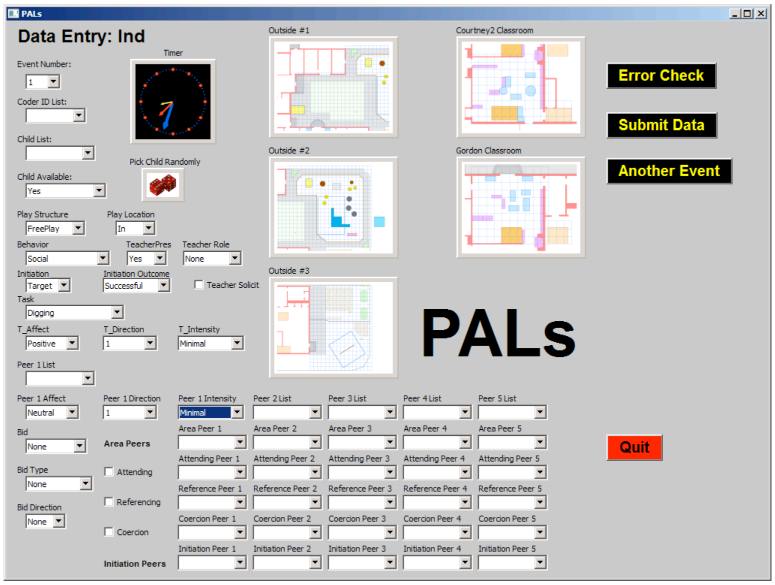


Figure 3.3: A map of the preschool structure[9]



Data Entry: Ind

Event Number: 1

Coder ID List:

Child List:

Child Available: Yes

Play Structure: FreePlay

Play Location: In

Behavior: Social

TeacherPres: Yes

Teacher Role: None

Initiation: Target

Initiation Outcome: Successful

□ Teacher Solicit

Task: Digging

T_Affect: Positive

T_Direction: 1

T_Intensity: Minimal

Peer 1 List:

Peer 1 Affect: Neutral

Peer 1 Direction: 1

Peer 1 Intensity: Minimal

Bid: None

Bid Type: None

Bid Direction: None

Area Peers

□ Attending

□ Referencing

□ Coercion

Initiation Peers

Peer 2 List	Peer 3 List	Peer 4 List	Peer 5 List	
Area Peer 1	Area Peer 2	Area Peer 3	Area Peer 4	Area Peer 5
Attending Peer 1	Attending Peer 2	Attending Peer 3	Attending Peer 4	Attending Peer 5
Reference Peer 1	Reference Peer 2	Reference Peer 3	Reference Peer 4	Reference Peer 5
Coercion Peer 1	Coercion Peer 2	Coercion Peer 3	Coercion Peer 4	Coercion Peer 5
Initiation Peer 1	Initiation Peer 2	Initiation Peer 3	Initiation Peer 4	Initiation Peer 5

Figure 3.4: A TabletPC based behavioral coding system: Coders use a digital pen to record preschool children's behavior in a GIS-based graphic user interface

[9] This map is reused from NSF project "Modeling time, space, and behavior: Combining ABM & GIS to create typologies of playgroup dynamics in preschool children" by William Griffin, Paul Torrens, Jennifer Fewell (2006-2011).

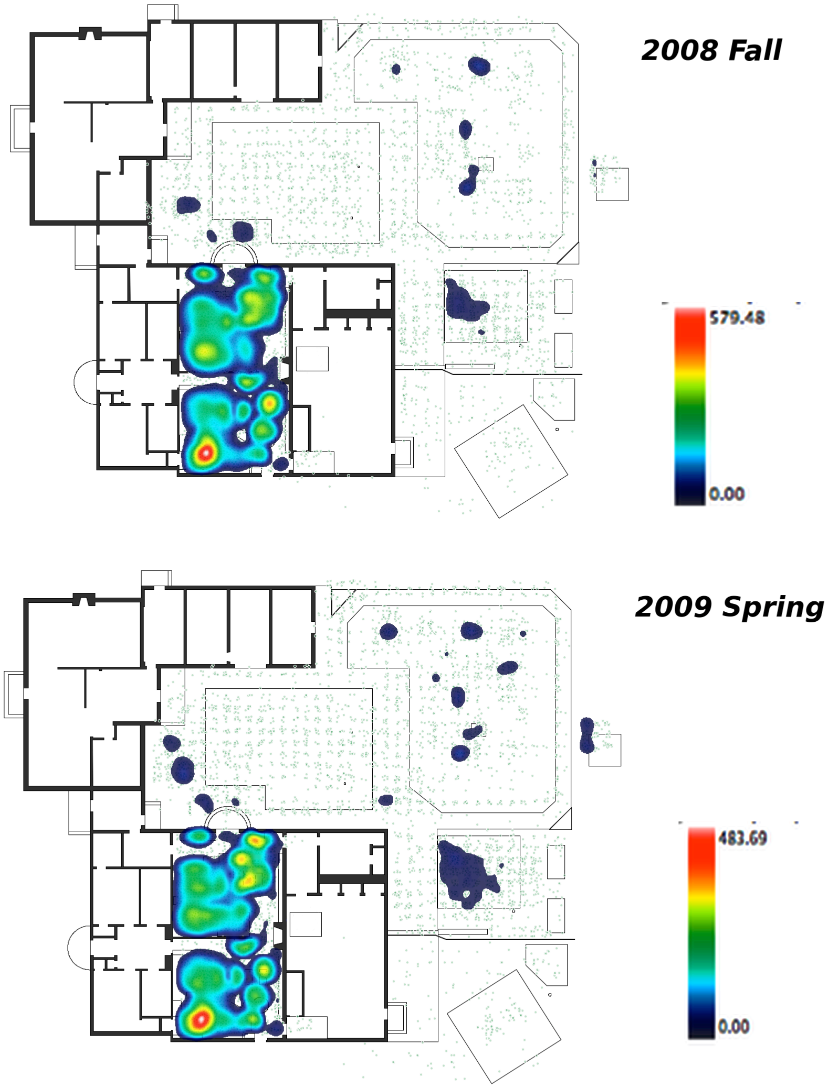


Figure 3.5: Kernel density maps of all observed indoor social activity for semester 2008 Fall (Upper), and 2009 Spring (Lower).

3.4.2 Case Study Applications

ESDA is used to explore social and spatial clusters of preschool children's behavior through the relationship of (1) physical places to children's social behaviors (spatial weights) and (2) the above-mentioned activity settings and children’s social structure (social weights).

To explore the spatial autocorrelation patterns, a global measure of spatial autocorrelation (Moran’s I index) is applied to diagnose the presence of spatial autocorrelation of children’s different behaviors for different gender in the study area. The results of standardized global Moran’s I indexes and their pseudo significance levels are shown in table 3.2, table 3.3 and summarized in figure 3.6. This research also tested local spatial autocorrelation by sweeping through the data, lattice-by-lattice and testing for autocorrelation around each lattice, with a Queen contiguity weights matrix. This allows us to examine the presence of “hotspot” relationships (high-high clusters) in which high frequencies of a variable were surrounded by similar neighbors in a statistically significant relationship, and “coldspot” relationships (low-low clusters) in areas with below-average values situated next to neighbors with below-average values. Interesting spatial patterns of children's behaviorand differences between boys and girls are shown in figure 3.7, 3.8 and 3.9.

To explore the patterns of spatial autocorrelation with social weights, global Moran’s I indices are measured to diagnose the presence of spatial autocorrelation of children’s 15 different activities in this social network by using social weights. To assess the sensitivity of the results to different social distance weights matrices, I chose *k* equal to 3, 4 and 5 for KNN weights. The results of standardized global Moran’s I indices and pseudo significance levels for children in two different classrooms are shown in table 3.4 and 3.5. To identify what activity is autocorrelated for which part of the social network, local spatial autocorrelation tests with social distance weights are applied to all 15 activities. The social distance at which the first quartile of all distance values is reached is selected experimentally as the bandwidth. The “hotspot” clusters may suggest a positive correlation between current activity settings and the underlying social network structure, while “coldspot” clusters may suggest a negative correlation (see figure 3.10 and 3.11). Although the results of spatial autocorrelation with social weights provide useful information, further research is still needed to address the question of which children interact on which activities.

3.4.3 Results

3.4.3.1 Spatial Patterns of Preschool Children’s Behavior

The results for the global Moran’s I test reveal strong positive spatial autocorrelation for all four behaviors indoor and outdoor (see table 3.2 and 3.3 in appendix, almost all behaviors over the entire study period are significant with pseudo-p values equal to 0.001). Comparing the Z values of Moran’s I tests for indoor and outdoor separately, we can see that non-social behavior (solitary) shows less spatial autocorrelation indoor (comparing to social behaviors indoor, see figure 3.6 upper, lighter color for solitary and darker color for rest 3 behaviors), while non-social behavior shows more spatial autocorrelation outdoor (comparing to social behaviors outdoor, see figure 3.6 lower). This could indicate that, when children played alone, they preferred to play outdoors. It is also interesting that, for both indoor and outdoor, the combined observations of social and teacher oriented behaviors shows more spatial autocorrelation than boy or girl only observations. This may indicate that, when boy and girl play together either socially or oriented by teachers, their activities are more associated with space than they play with same-sex friends.

The related LISA maps indicate that indoor environmental settings such as the study corner and bookshelf (see “hotspots” located indoor in figure 3.7) are positively associated with children’s social behavior: children are using these resources for socialization. Meanwhile, significant “coldspots” are mostly located in corridors and corner areas. An interesting phenomenon is that table, which should be used for socialization, showed more teacher interaction, solitary and parallel hotspots and less social peer interaction “hotspots”. In outdoor environment, play resource settings such as sand box, tent area and “play house” area in front of classroom door host most “hotspots” that are positively correlated with children’s social behaviors.

The outdoor environment settings such as the sandbox, tent area and area near “play house” (see “hotspots” located outdoors for solitary behavior in figure 3.7) are also associated with solitary behavior. This indicates that such environmental settings geared towards both individual and group activities. The characteristics of outdoor environmental settings, such as the large sandbox and area near “play house) that can contains children with various behaviors) and large tent area that servers different play resource (e.g. self-task designed climber and cooperation designed slider) also match the above results. This may lead to more outdoor social activities, which matches the results that more “hotspots” were detected.

The analysis of LISA also indicates that spatial patterns of children’s behavior observed in this study were not necessarily constant over time. By comparing the LISA maps of the 4 different behaviors across 2 different semesters (see figure 3.7), the overall change of clusters over time for all activities shows that children played in different areas at different times. For example, the indoor clusters vary at each semester for all behaviors. This could be because of the change of classroom settings in each semester (e.g. reorganized furniture in classroom), and children’s major activities are closely associated with these classroom settings. The clusters that are at fixed positions over time are associated with children playing in stationary environmental settings. For example, “high-high” clusters are always observed in the outdoor sand box for the 4 different types of behavior, which indicates that the sandbox is important for various activities. We can also discover the gradual development of children’s parallel behavior in the outdoor environment over time from the LISA maps in 2 time periods. These maps indicate that the hotspots of parallel behavior grow outdoors and settle in several fixed places. This could suggest that environmental settings contribute to developing children's social behavior.

To examine the sex differences of children’s behavior regarding specific environmental settings, spatial analysis is also applied separately to behavioral data of boys and girls. Standardized global Moran’s I tests can be found in table 3.4 and LISA maps of boys' and girls' 4 different behaviors are displayed in figure 3.7 and 3.8. One interesting difference between boys’ and girls’ “hotspots” of “solitary” behavior can be observed indoors: in every semester, when boys formed hotspots in the free space of one classroom, girls formed hotspots around tables in another classroom. The position of boys' hotspots is opposite to girls' hotspots. The same separation pattern can be observed in "teacher oriented" behavior: when boys formed hotspots around the teacher in one classroom, girls did not do the same, and vice versa. This suggests that there are distinct spatial differences between boys and girls when they play by themselves or with the teacher. These separation patterns indicate that children prefer to be around other children of the same gender for some non-social activities.

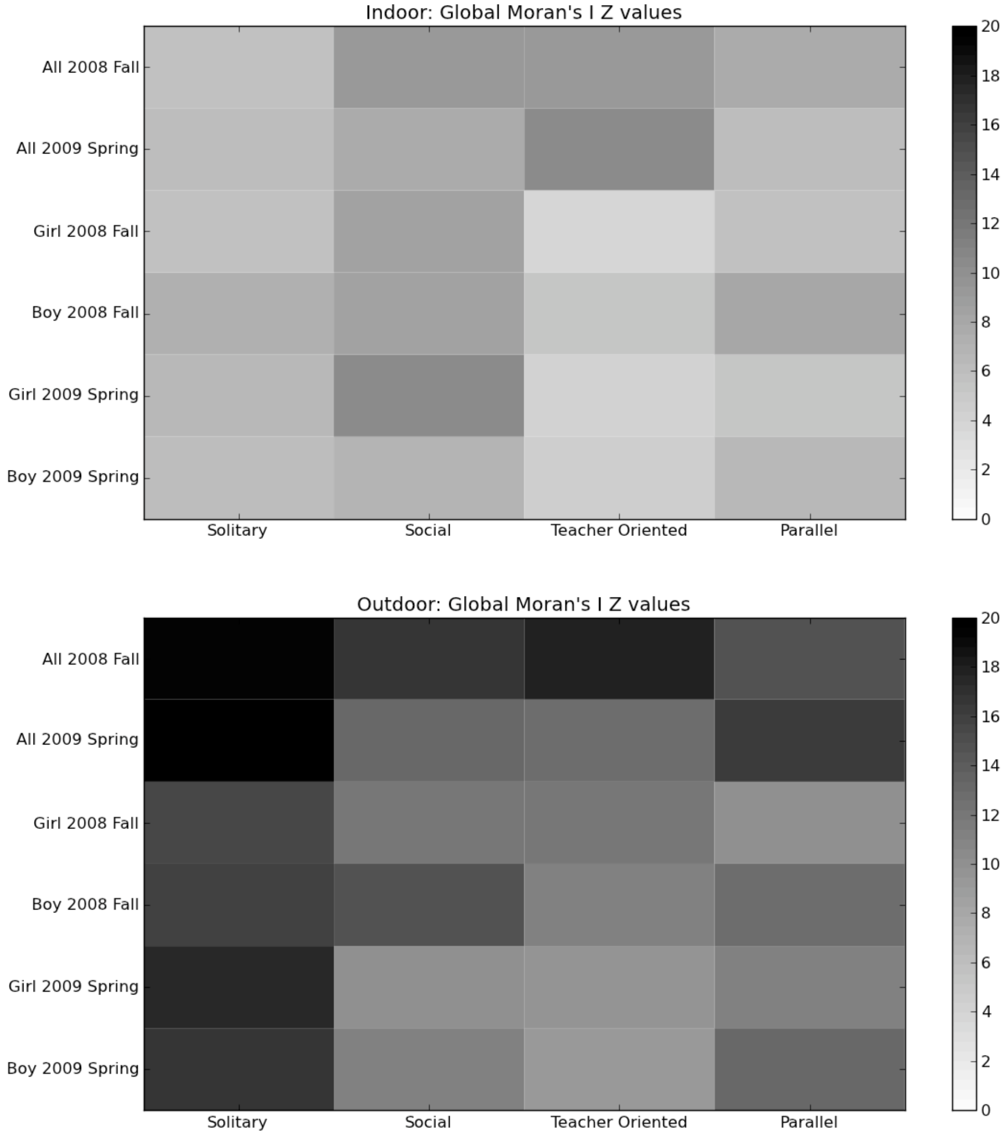


Figure 3.6: The plots of Global Moran's I Z values of four behaviors in two semesters from observations of mixed gender, boy only and girl only. (Upper plot is for indoor Z values of Global Moran's I tests, lower plot is for outdoor Z values of Global Moran's I test.)

Unlike the distinct separation of spatial clustering in "solitary" and "teacher oriented" behavior, "social" and "parallel" social behaviors exhibit more overlap in hotspots formed by both boys and girls in an indoor environment. This overlap in spatial clusters of social behavior suggests that boy and girl interactions are part of children’s social behavior. It is also interesting that there are fewer overlapping and more separate hotspots in social behavior than in indirect peer interacting behavior (“parallel”). This could mean that children played more with same-sex playmates but they were learning to play with opposite-sex playmates. However, in outdoor environments, such distinct differences cannot be identified. Boys appear to favor more outdoor areas than girls since more outdoor hotspots can be observed for boys than girls. Still, they shared a lot of hotspots for 4 different behaviors. This could be because the outdoor space is larger than the indoor space, so that children can maintain their private space easier outdoors.

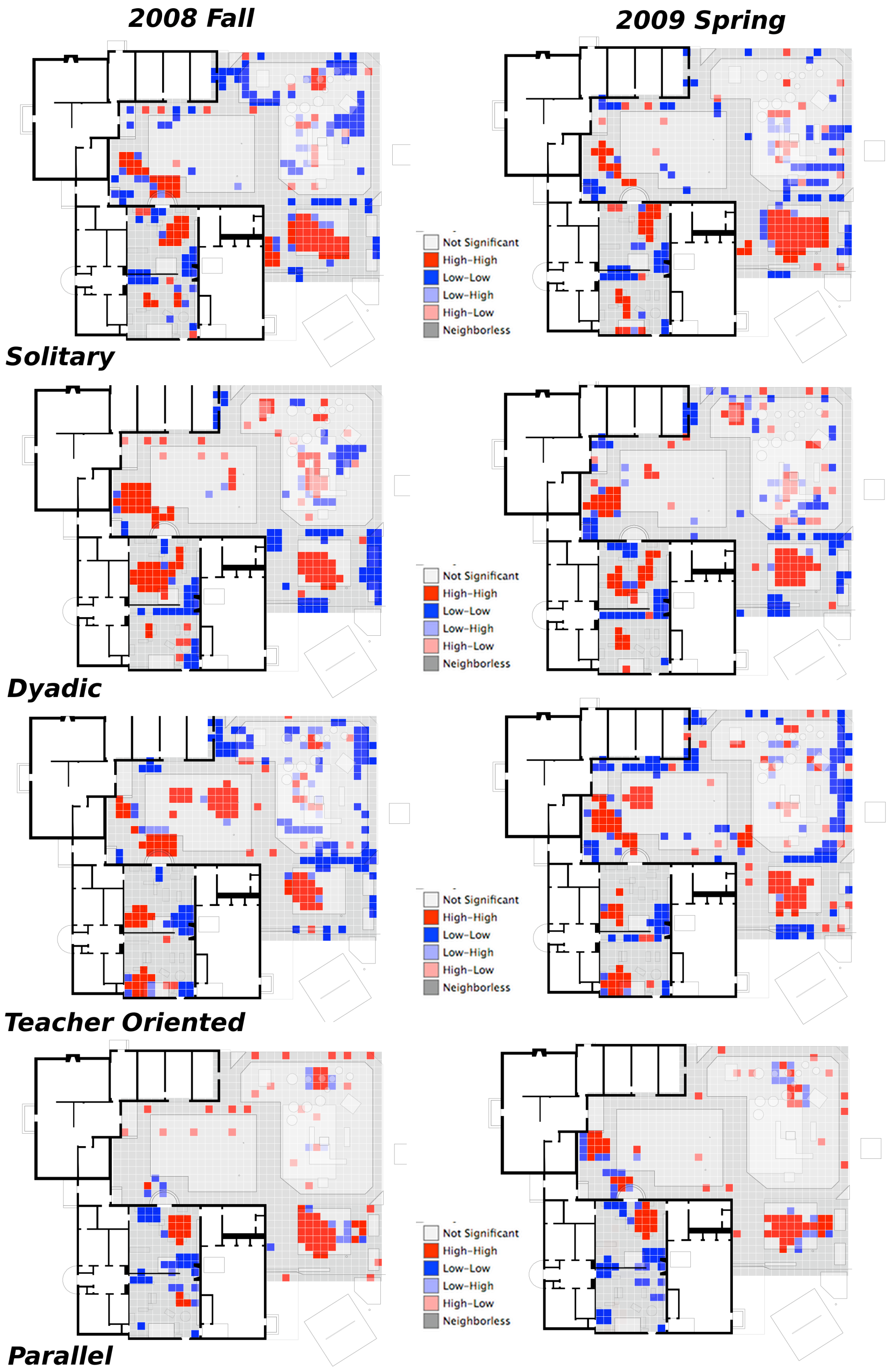


Figure 3.7: LISA maps by type of behavior and date, for all children in semester 2008 Fall and 2009 Spring (only cluster cores are shown).

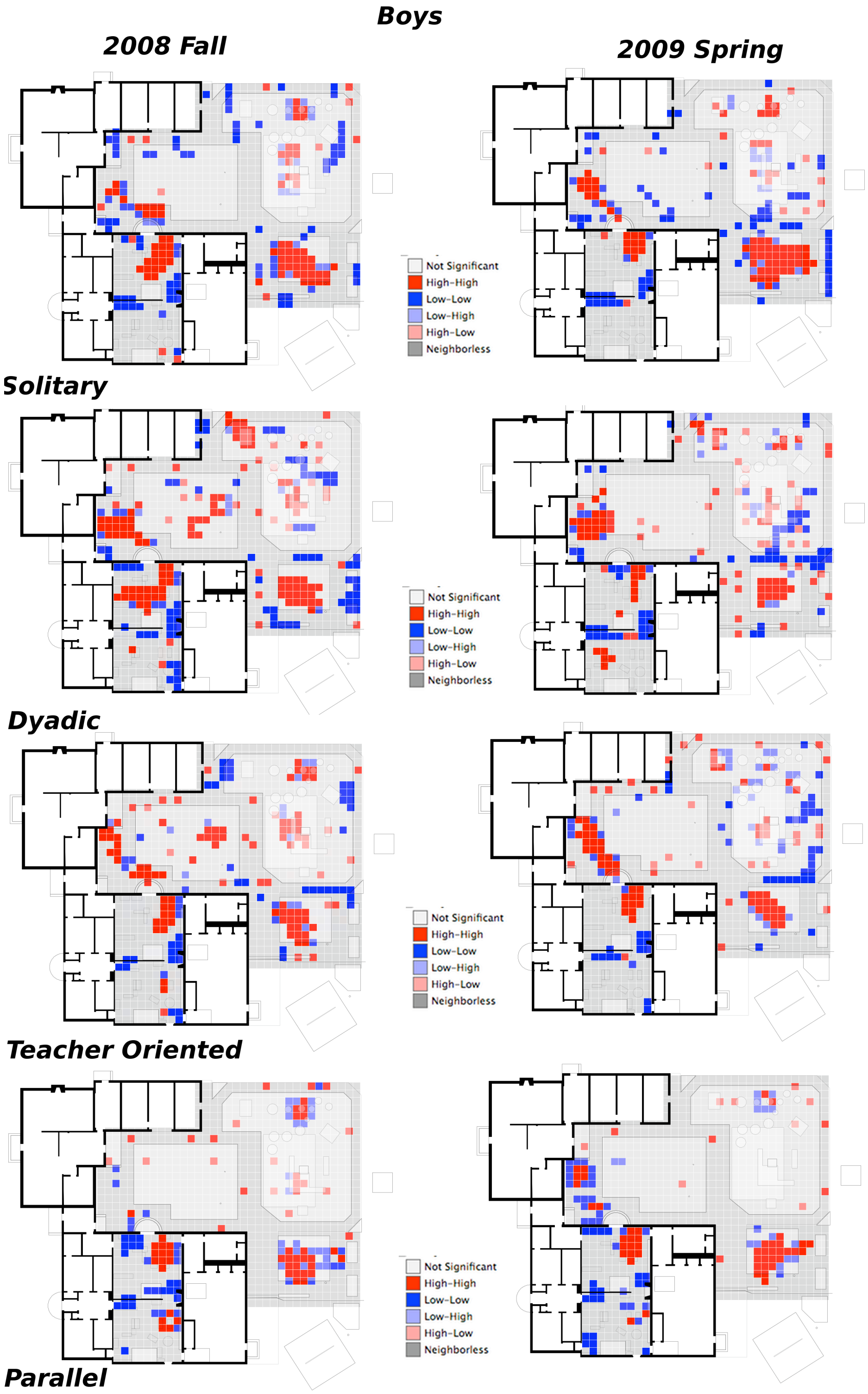


Figure 3.8: LISA maps by type of behavior and date, for boys only in semester 2008 Fall and 2009 Spring (only cluster cores are shown).

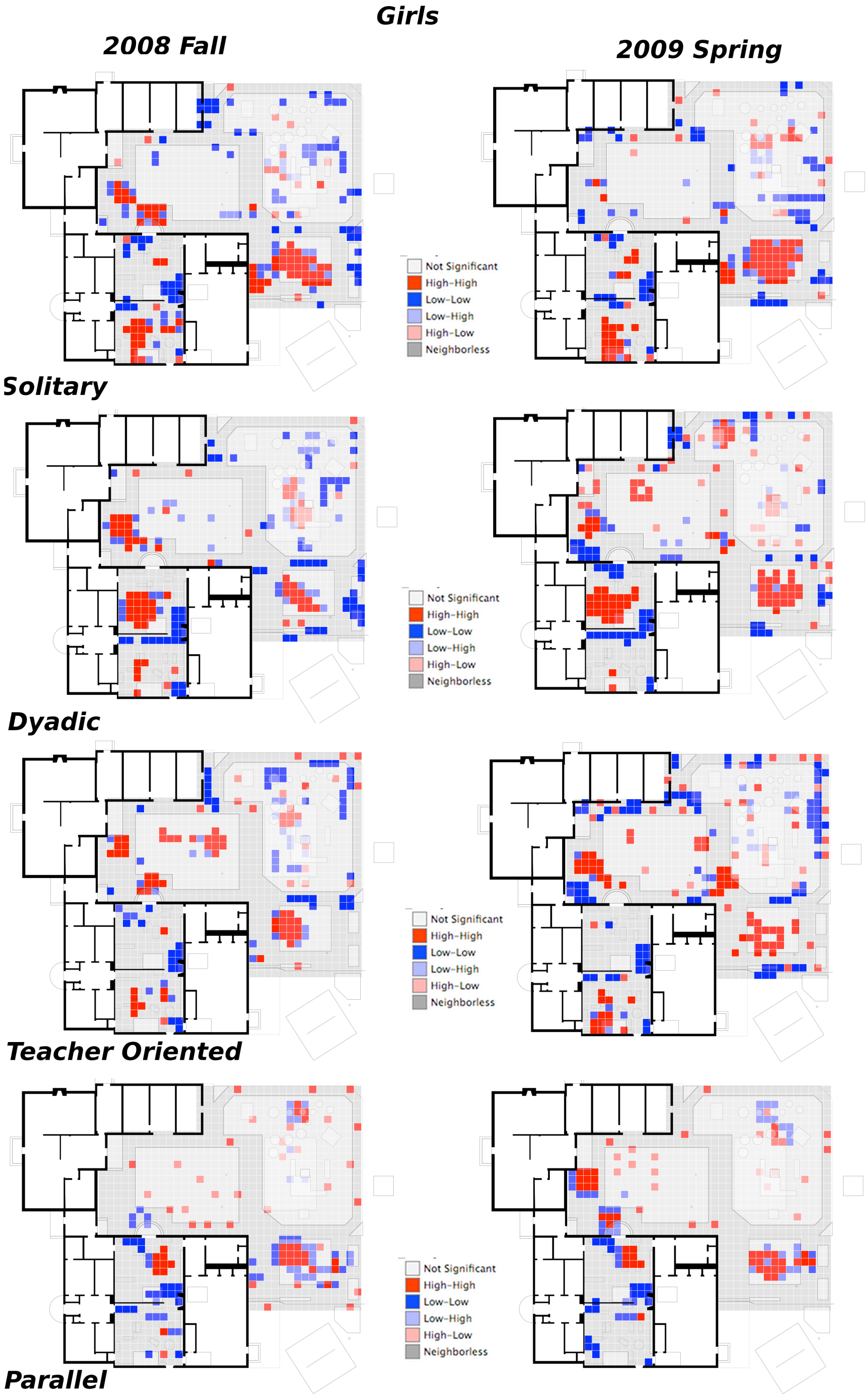


Figure 3.9: LISA maps by type of behavior and date, for girls only in semester 2008 Fall and 2009 Spring (only cluster cores are shown).

3.4.3.2 Spatial Autocorrelation of Preschool Children's Activities

The results for the global Moran's I test of spatial autocorrelation with social weights reveal the positive to negative autocorrelation of children's activities. Spatial autocorrelation with social weights differs by group of children and also varies across time. The significant standardized spatial autocorrelation test results with *p*-value less than 0.05 for 15 tasks in two different classrooms (classroom 1 and 2) and two different time periods (2008 Fall and 2009 Spring semester) are highlighted in table 3.4 and 3.5.

In general, children's propensities of specific tasks (activities) have a positive correlation with their social clusters in every semester. For example, in classroom 1, significant clustering patterns can be identified at "Figure Play", "Music/Sing" and "Sensory Play" (sorted from high to low significant spatial autocorrelation) in the spring Fall 2008; while in same semester, in classroom 2, significant clustering patterns can be identified at "Molding", "Manipulatives", "Art" and "Pretend Play". This means that children who are friends (frequently play together) also have same preference of these activities. Whether children played together during engaging in these activities would require further investigation. For example, by checking the time series of children's activities and interactions together at micro-scale, we can get the details about if these activities happened at same space and time between children.

Some tasks, such as "Pretend Play", "Figure Play" and "Sensory Play", are designed to be accomplished with a lot of social cooperation and communication for several children. Therefore, children's preference for these activities may positively associated with forming their social groups. However, other tasks, such as “Art”, “Manipulatives” and “Language Arts”, seem to be more individual-based activities but are also have positive autocorrelation with social weights. This result may demonstrate that having similar taste on some individual oriented activities is positively correlation with children’s friendship.

However, such correlation is not fixed for all children and at all the time. For example, “Figure Play”, “Music/Sing” and “Sensory Play” exhibit positive correlation among friends in classroom 1 in 2008 Fall, but in 2009 Spring, “Art” replaces “Figure Play”. For children in classroom 2, the patterns are totally different: socially close children are associated with having similar propensity in “Art”, “Manipulative”, and “Pretend Play” etc. in 2008 Fall, and further changes to “Art”, “Board Games”, “Digging sand” etc. in 2009 Spring. An interesting result, which requires further investigation, is that the “Music/Sing” activity has positive correlation among socially close children in classroom 1, but has negative correlation for children in classroom 2 in the 2008 Fall semester.

To further examine where are the “significant” activity clusters for which socially close children, the LISAs with social weights provide more detail information. The results of the top 3 activities with positive spatial autocorrelation among socially close children for each classroom are shown in LISA maps (see figure 3.10 and 3.11). The LISA clusters in figure 3.10 provide information about which children form “high-high” clusters for engaging a specific activity. For example, the “Figure Play” LISA cluster graph in semester Fall 2008 identifies “hotspots” among children #23, #232, #233and #299[10] in classroom 1. These children are socially close in social network space based on their interactions, and they all attend "Figure Play" more frequently than other children at a statistically significant level. This shows a similar propensity of doing “Figure Play” exists among these socially close children. Similar “high-high” clusters can be found in “Music/Sing” among child #110, #116, #250 and #274, and in “Sensory Play” among child #110, #250, #274 and #140. It is interesting that some playmates appear as “high-high” clusters in several tasks, showing a strong shared preference for similar activities. For example, child #274, #250 and #110 in classroom 1 form clusters in both “Music/Sing” and “Sensory Play” in the 2008 Fall semester (see figure 3.10).

It is also worth noting that “low-low” clusters in blue can be used to identify children who are socially connected but infrequently engage in a particular activity. For example, in 2008 Fall semester, child #137 and #135 playing “Molding” in classroom #2; in Spring 2009, child #121 playing “Music/Sing” in classroom 1 and child #27 playing “Art” in classroom 2 (see figure 3.10 and 3.11). These “low-low” clusters may indicate that children with their socially close friends were rarely participating or not interested in these activities. Meanwhile, “low-high” clusters represent that some children who play a lot with their friends but have opposite engagement frequencies or preferences in regards to a particular activity. For example, in the “Figure Play” LISA graph

[10] The number in each node represents the identifier (ID) of a child. The arbitrary ID has been randomized for protecting the children’s privacy.

(2008 Fall in classroom 1), child #121, who is located at the "margin" (means this is a less social child) of the social network, is a significant "low-high" cluster, which means this child is not interested in "Figure Play" activity while his/her friends participated significant frequently in this activity (see the "hotspots" around).

Moreover, sex differences in activity engagement can be observed from the results of the LISA graphs. In this case study, all girls were set to have even ID numbers and all boys were set to have odd ID numbers. It is interesting that all identified "high-high" clusters in all significant tasks in both classrooms are either "boys' cluster" or "girls' cluster" (see figure 3.10 and 3.11). Why this happens requires further investigation, but this may suggest a stratification of activity engagement by sex.

Table 3.4

Testing Results for Global Spatial Autocorrelation with Social Weights using the Moran's I Statistics (Classroom 1). The Values with p-value <= 0.05 are Highlighted in Gray Color.

Classroom1	2008 Fall			2009 Spring		
	Global Moran's I	Pseudo P-value	Z-value	Global Moran's I	Pseudo P-value	Z-value
Art	-0.120	0.338	-0.360	0.252	0.023	2.063
Board games	-0.181	0.14	-0.748	-0.095	0.402	-0.272
Digging sand	0.092	0.31	1.085	-0.147	0.239	-0.627
Figure play	0.366	0.003	2.759	-0.044	0.463	0.115
Language arts	0.079	0.153	0.842	-0.011	0.39	0.265
Large motor	0.015	0.319	0.480	-0.153	0.261	0.659
Manipulatives	0.183	0.046	1.682	-0.064	0.49	-0.107
Math/science	-0.082	0.499	-0.123	-0.050	0.479	0.043
Molding	-0.056	0.477	0.069	-0.068	0.444	-0.109
Music/sing	0.361	0.006	3.133	0.400	0.004	3.327
Physical games	-0.166	0.249	-0.731	0.023	0.236	0.748
Pretend play	0.082	0.144	1.092	-0.055	0.454	0.033
Sensory play	0.207	0.042	1.744	0.180	0.037	1.769
Talk	-0.105	0.412	-0.287	-0.118	0.333	-0.441
Walking	-0.005	0.319	0.415	-0.181	0.202	-0.858

Table 3.5

Testing Results for Global Spatial Autocorrelation with Social Weights using the Moran's I Statistics (Classroom 2). The Values with p-value <= 0.05 are Highlighted in Gray Color)

Classroom 2	2008 Fall			2009 Spring		
	Global Moran's I	Pseudo P-value	Z-value	Global Moran's I	Pseudo P-value	Z-value
Art	0.231	0.003	2.968	0.816	0.001	5.983
Board games	-0.022	0.388	0.247	0.473	0.003	3.625
Digging sand	0.019	0.285	0.532	0.221	0.042	1.904
Figure play	0.110	0.135	1.105	0.094	0.165	0.952
Language arts	-0.144	0.259	-0.631	-0.258	0.059	-1.448
Large motor	-0.225	0.187	-0.953	0.148	0.077	1.568
Manipulatives	0.319	0.012	2.333	-0.079	0.459	-0.130
Math/science	-0.139	0.338	-0.523	-0.014	0.371	0.239
Molding	0.330	0.018	2.486	-0.032	0.4	0.209
Music/sing	-0.401	0.015	-2.133	0.044	0.231	0.639
Physical games	0.049	0.251	0.640	0.222	0.025	2.074
Pretend play	0.306	0.017	2.007	0.344	0.01	2.682
Sensory play	0.049	0.204	0.835	-0.068	0.487	-0.057
Talk	0.138	0.12	1.135	0.408	0.008	2.976
Walking	0.191	0.078	1.511	0.160	0.065	1.466

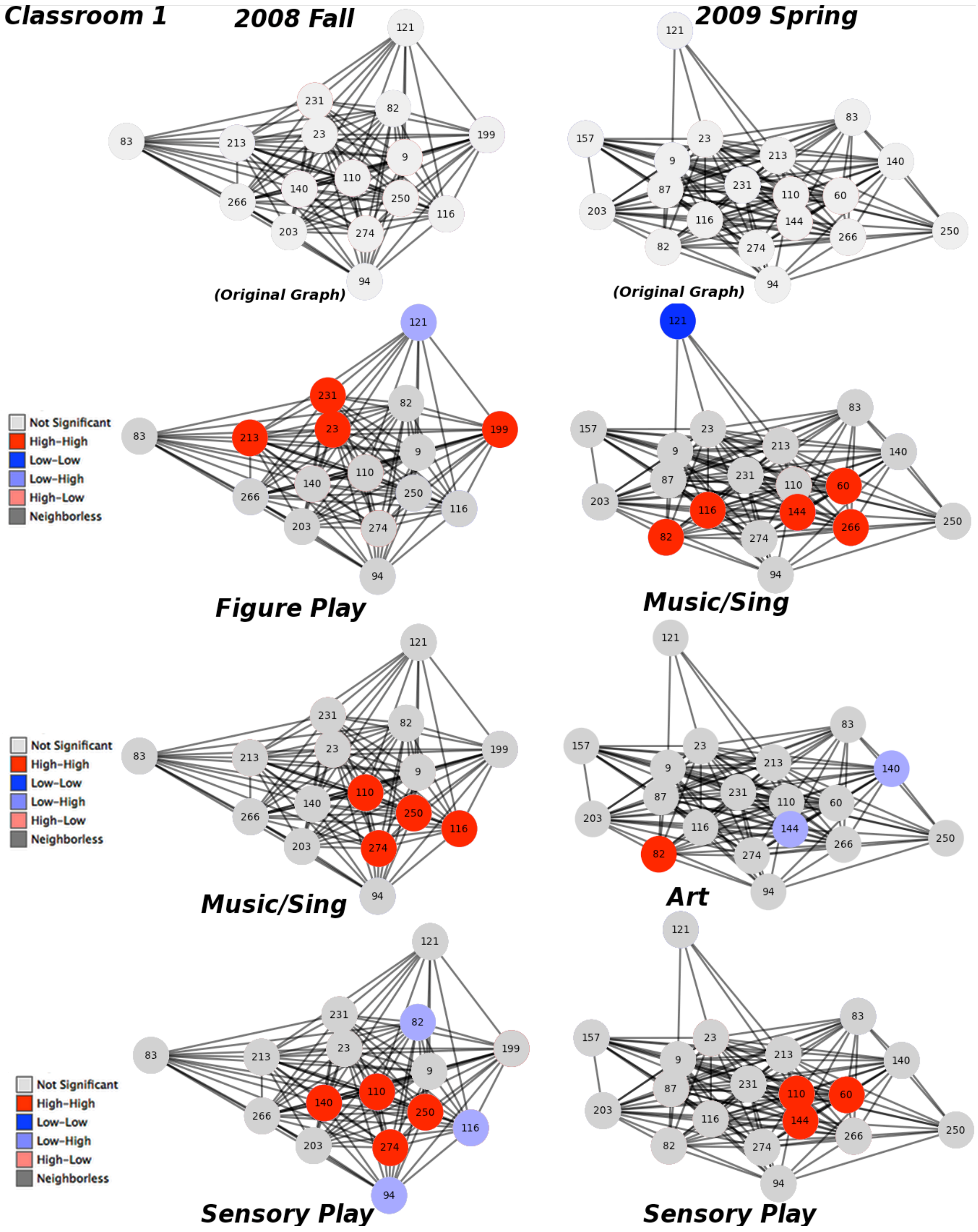


Figure 3.10: Top 3 local social network autocorrelation graphs by type of activities in classroom 1 in 2008 Fall and 2009 Spring semesters

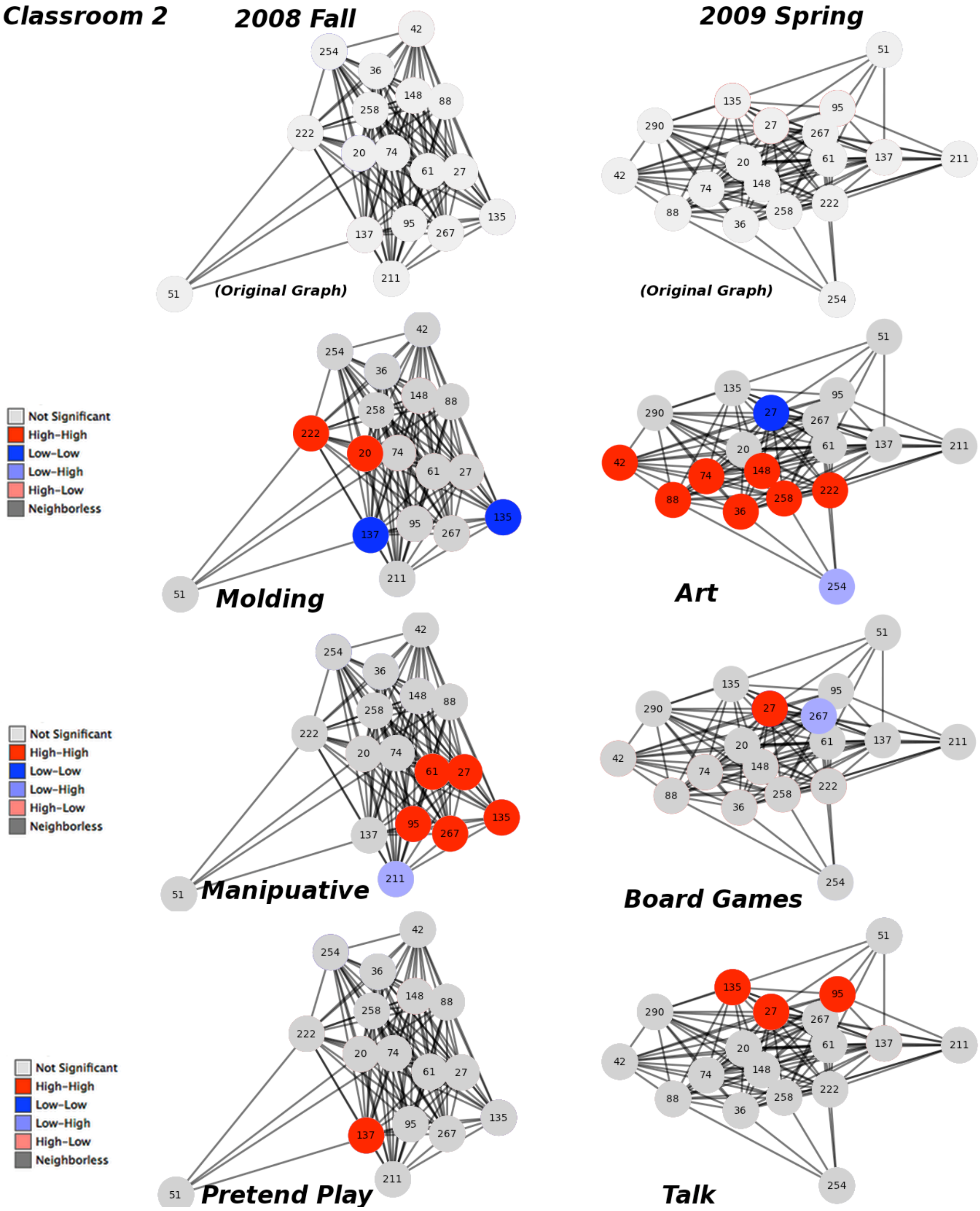


Figure 3.11: Top 3 local social network autocorrelation graphs by type of activities in classroom 2 in 2008 Fall and 2009 Spring semesters

3.5 Conclusion

In this essay, by integrating GIS, spatial analysis and social network visualization techniques, I applied ESDA with spatial and social weights to examine group-level spatial and social patterns of preschool children's behavior. I use this methodology along with contemporary observation methods to demonstrate that the spatial and social patterns discovered from spatiotemporal micro-social data are useful for studying the socialization in preschool children at aggregate levels. Spatial autocorrelation with spatial weights enables us to explore the association between preschool's environment and children's play behaviors. Spatial autocorrelation with social weights enables us to explore the association between preschool children's preference on school activities and children's social relationship by examining their correlations in a social network context. This new perspective provides a better understanding of the development of children's social behavior by answering questions related to preschool children's research such as "who is playing with each other, how do they play and where do they play".

First, the spatial autocorrelation analysis of children's socio-spatial behavior indicated that children clustered their different types of daily activities in specific locations with special resources. Some resources are associated with particular behaviors (e.g. desks for solitary and parallel behavior), while other resources are places for all types of behaviors (e.g. sand box). These clustering patterns also changed over time and varied based on different groups of children. Different socio-spatial clusters formed at different time periods in this study.

Moreover, these clustering patterns geographically differentiate boys from girls for certain behaviors. In particular, boys with solitary and teacher oriented behaviors formed clusters in one classroom while girls formed clusters in another classroom.

Second, from the analysis of children's social behavior using spatial autocorrelation with social weights, the activity settings for preschool children also have a significant correlation with children's social networks. This approach can statistically identify which tasks (activities) assigned to children have a positive correlation with social ties among children. Examining the LISA with social weights for these activities reveal which children are socially close and spend more time on specific activities (although not necessarily with each other). Activity outliers among playmates are identified from LISA clusters ("low-low" and "low-high"). These tasks also have different spatial correlations in different time periods. Sex differences in activity engagement are observed from the results of LISA graphs.

In summary, a socio-spatial approach as applied in this essay introduces analysis methods from geography and spatial analysis by investigating the relationship between environmental settings and preschool children's social behavior. Although I use preschool children to illustrate the use of these methods, this methodology and toolkit can be more broadly applied in the social sciences for studying human social behavior and human socialization. However, this approach also has limitations. For one, it does not allow us to directly test the relationship between activities and socialization. Further, selecting an appropriate geographic, social and temporal scale is difficult because there are many scale options and these choices directly impact the final results. For example, choosing semester as a time scale will involve more children (considering preschool children are an unstable group) and more observations than using month as a time scale. Therefore, choosing month as time scale could lead different social weights, which will impact the results of spatial autocorrelation test. Questions like, what is the best size of spatial observation unit for spatial analysis, is this unit suitable for both indoor and outdoor, or what is the best size of time period for studying children's socialization, need further investigations. Other issues like how to define appropriate social weights are important for a fruitful analysis of spatial autocorrelation with social weights. There are several discussions about how to define social distance (see section 3.3.2.1) but none of them are targeting preschool children and applying LISAs with social weights.

3.6 References

Aizer, A. & J. Currie (2004) Networks or neighborhoods? Correlations in the use of publicly-funded maternity care in California. Journal of Public Economics, 88, 2573-2585.

Anselin, L. (1994) Exploratory spatial data analysis and geographic information systems. New Tools for Spatial Analysis, 45-54.

--- (1995) Local indicators of spatial association—LISA. Geographical Analysis, 27, 93-115.

--- (2010) Thirty years of spatial econometrics. Papers in Regional Science, 89, 3-25.

Anselin, L., R. J. G. M. Florax & S. J. Rey. 2004. Advances in spatial econometrics: methodology, tools and applications. Berlin: Springer-Verlag.

Anselin, L., S. Sridharan & S. Gholston (2007) Using exploratory spatial data analysis to leverage social indicator databases: the discovery of interesting patterns. Social Indicators Research, 82, 287-309.

Barbour, A. C. (1999) The impact of playground design on the play behaviors of children with differing levels of physical competence. Early Childhood Research Quarterly, 14, 75-98.

Bertrand, M., E. F. P. Luttmer & S. Mullainathan (1998) Network effects and welfare cultures. Quarterly Journal of Economics, 115, 1019-1055.

Black, W. R. (1992) Network autocorrelation in transport network and flow systems. Geographical Analysis, 24, 207-222.

Bogardus, E. S. (1925) Measuring social distance. Journal of Applied Sociology, 9, 299-308.

Boots, B. N. & P. S. Kanaroglou (1988) Incorporating the effects of spatial structure in discrete choice models of migration. Journal of Regional Science, 28, 495-510.

Brewer, M. B., H. K. Ho, J. Y. Lee & N. Miller (1987) Social identity and social distance among Hong Kong schoolchildren. Personality and Social Psychology Bulletin, 13, 156.

Conley, T. G. & G. Topa (2002) Socio-economic distance and spatial patterns in unemployment. Journal of Applied Econometrics, 17, 303-327.

Epstein, J. M. 2006. Generative social science: Studies in agent-based computational modeling. Princeton: Princeton University Press.

Farber, S., A. Páez & E. Volz (2009) Topology and dependency tests in spatial and network autoregressive models. Geographical Analysis, 41, 158-180.

Fishbein, H. D. & S. Imai (1993) Preschoolers select playmates on the basis of gender and race. Journal of Applied Developmental Psychology, 14, 303-316.

Fotheringham, A. S., M. E. Charlton & C. Brunsdon (2001) Spatial variations in school performance: A local analysis using geographically weighted regression. Geographical and Environmental Modelling, 5, 43-66.

Gresham, F. M., G. Sugai & R. H. Horner (2001) Interpreting outcomes of social skills training for students with high-incidence disabilities. Exceptional Children, 67, 331-344.

Griffin, W. A., S. K. Schmidt, A. Nara, P. M. Torrens, J. H. Fewell & C. Sechler. 2007. Integrating ABM and GIS to model typologies of playgroup dynamics in preschool children. In Agent 2007, ed. C. M. D. S. M. North, 17-24. Evanston, IL: Argonne National Labs and the University of Chicago.

Gutierrez Jr, A., M. N. Hale, K. Gossens-Archuleta & V. Sobrino-Sanchez (2007) Evaluating the social behavior of preschool children with autism in an inclusive playground setting. International Journal of Special Education, 22, 5.

Holt, L. (2007) Children's sociospatial (re) production of disability within primary school playgrounds. Environment and Planning D, 25, 783.

Leenders, R. T. A. J. (2002) Modeling social influence through network autocorrelation: constructing the weight matrix. Social Networks, 24, 21-47.

Leman, P. J. & V. L. Lam (2008) The influence of race and gender on children’s conversations and playmate choices. Child Development, 79, 1329-1343.

Marsden, P. V. & N. E. Friedkin (1993) Network studies of social influence. Sociological Methods & Research, 22, 127-151.

Messner, S. F. & L. Anselin (2004) Spatial analyses of homicide with areal data. Spatially Integrated Social Science, 127-144.

Oden, S. & S. R. Asher (1977) Coaching children in social skills for friendship making. Child Development, 495-506.

Parker, R. N. & E. K. Asencio. 2008. GIS and spatial analysis for the social sciences: coding, mapping and modeling. New York: Routledge.

Plowman, L. & C. Stephen (2005) Children, play, and computers in pre-school education. British Journal of Educational Technology, 36, 145-157.

Quilitch, H. R. & T. R. Risley (1973) The effects of play materials on social play. Journal of Applied Behavior Analysis, 6, 573.

Radil, S. M., C. Flint & G. E. Tita (2010) Spatializing social networks: Using social network analysis to investigate geographies of gang rivalry, territoriality, and violence in Los Angeles. Annals of the Association of American Geographers, 100, 307-326.

Rogers, C. S. & J. K. Sawyers. 1988. Play in the lives of children. Washington, DC: National Association for the Education of Young Children.

Rose, A. J. & K. D. Rudolph (2006) A review of sex differences in peer relationship processes: Potential trade-offs for the emotional and behavioral development of girls and boys. Psychological Bulletin, 132, 98.

Santos, A. J. & L. T. Wingegar (1999) Child social ethology and peer relations: a developmental review of methodology and findings. Acta Ethologica, 2(1), 1-11.

Santos, A. J., B. E. Vaughn & K. K. Bost (2008) Specifying socicla structures in preschool classrooms: descriptive and functional distinctions between affiliative subgroups. Acta Ethologica, 11(2), 101-113.

Sayer, A. 1992. Method in social science: A realist approach. London: Hutchinson.

Shores, R. E., S. L. Jack, P. L. Gunter, D. N. Ellis, T. J. DeBriere & J. H. Wehby (1993) Classroom interactions of children with behavior disorders. Journal of Emotional and Behavioral Disorders, 1, 27.

Tita, G. E. & S. M. Radil (2011) Spatializing the social networks of gangs to explore patterns of violence. Journal of Quantitative Criminology, 1-25.

Torrens, P., X. Li & W. A. Griffin (2011) Building agent-based walking models by machine-learning on diverse databases of space-time trajectory samples. Transactions in GIS, 15, 67-94.

Turner, P. J. (1991) Relations between attachment, gender, and behavior with peers in preschool. Child Development, 62, 1475-1488.

Vaughn, B. E., T. N. Colvin, M. R. Azria, L. Caya & L. Krzysik (2001) Dyadic analyses of friendship in a sample of preschool-age children attending head start: correspondence between measures and implications for social competence. Child Development, 72(3), 862-78.

Whiting, B. B. & C. P. Edwards. 1992. Children of different worlds: The formation of social behavior. Cambridge, MA: Harvard University Press.

Yamada, I. & J. C. Thill (2007) Local Indicators of Network‐Constrained Clusters in Spatial Point Patterns. Geographical Analysis, 39, 268-292.

APPENDIX B

ADDITIONAL GLOBAL MORAN'S I TEST RESULTS

Table 3.2

Testing Results for Global Spatial Autocorrelation of Children’s Spatial Behaviors using the Moran’s *I* Statistic (Queen Weights Matrix)

		Indoor			Outdoor		
		Moran’s I	Pseudo P-value	z-value	Moran’s I	Pseudo P-value	z-value
Teacher interaction	2008 Fall	0.386	0.001	9.034	0.338	0.001	17.957
	2009 Spring	0.410	0.001	10.257	0.234	0.001	12.672
Social interaction	2008 Fall	0.385	0.001	9.312	0.317	0.001	16.818
	2009 Spring	0.337	0.001	7.848	0.242	0.001	13.132
Parallel Interaction	2008 Fall	0.305	0.001	7.724	0.281	0.001	14.859
	2009 Spring	0.254	0.001	6.300	0.304	0.001	16.484
Solitary Behavior	2008 Fall	0.252	0.001	5.917	0.355	0.001	19.411
	2009 Spring	0.257	0.001	6.200	0.372	0.001	20.070

Table 3.3

Testing Results for Global Spatial Autocorrelation of Boys' and Girls' Spatial Behaviors using the Moran’s I Statistic (Queen Weights Matrix)

			Indoor			Outdoor		
			Moran’s I	Pseudo p-value	z-value	Moran’s I	Pseudo p-value	z-value
Teacher interaction	2008 Fall	Boy	0.219	0.001	5.222	0.204	0.001	11.058
		Girl	0.163	0.002	3.932	0.217	0.001	11.815
	2009 Spring	Boy	0.186	0.001	4.506	0.167	0.001	9.136
		Girl	0.179	0.001	4.216	0.177	0.001	9.703
Social interaction	2008 Fall	Boy	0.363	0.001	8.374	0.274	0.001	14.642
		Girl	0.362	0.001	8.515	0.222	0.001	12.052
	2009 Spring	Boy	0.272	0.001	6.855	0.212	0.001	11.119
		Girl	0.433	0.001	10.401	0.180	0.001	9.899
Parallel Interaction	2008 Fall	Boy	0.340	0.001	8.137	0.223	0.001	12.738
		Girl	0.245	0.001	5.915	0.186	0.001	9.967
	2009 Spring	Boy	0.271	0.001	6.627	0.242	0.001	13.342
		Girl	0.214	0.001	5.324	0.214	0.001	11.031
Solitary Behavior	2008 Fall	Boy	0.296	0.001	7.220	0.288	0.001	15.786
		Girl	0.247	0.001	5.810	0.278	0.001	15.371
	2009 Spring	Boy	0.253	0.001	6.143	0.307	0.001	16.694
		Girl	0.283	0.001	6.416	0.331	0.001	17.437

Conclusion

To address the research challenge of discovering useful patterns and knowledge in increasingly ubiquitous, large-scale, electronically collected, spatiotemporal activity data, this dissertation collects three different new types of spatiotemporal data and targets three different research objectives: (1) using the spatiotemporal information embedded in massive online geo-tagged photos to build an intelligent travel trip plan system for automatically recommending multi-day and multi-stay travel itineraries to travelers, (2) training a classification model to automatically determine the movement type of unknown trajectories from massive crowd sourced GPS trajectories, and (3) discovering the group-level spatial and social patterns of preschool children's playing behaviors for studying the socialization in preschool children at aggregate levels from spatiotemporal micro-social data collected using TabletPCs.

Results of the three objectives in this dissertation have led to the development of the methodological framework for spatiotemporal data mining, analysis and visualization of new forms of human activity data.

For the first objective, the first essay develops an intelligent travel trip plan system based on discovered attractions, travel patterns, and traveling graph models from geo-tagged photos. Extending existing data mining, spatial optimization and geovisualization techniques, this system can automatically recommend multi-day and multi-stay travel itinerary that generates the approximate maximum attractiveness score for inexperienced travelers, who only know the travel origination and destination, and available time. The generated travel itinerary includes a text description of when to start the trip, where to visit, how long to stay and how long to drive to the next attraction for every travel day, as well as driving directions and a related map to help tourists travel.

The second essay provides a new machine-learned classification model for automatically determining the movement type of unknown trajectories. This model introduces two new types of complexity measures as new features for classifying movements: the geometric complexity measures of trajectories based on Fractal Dimensions, and structural complexity measures of movement parameters based on Approximate Entropy. These two types of complexity measures highlight both general geometric characteristics and the subtle changes of movement parameters that exist in different moving trajectories in the classification model. The overall 85.4% average accuracy of prediction outperforms the existing state-of-the-art classification model, and demonstrates the applicability of this classification model.

For the third objective, the third essay applies ESDA with spatial and social weights along with GIS, spatial analysis and social network analysis techniques to micro-social data to examine group-level spatial and social patterns of preschool children's play behaviors. Spatial autocorrelation with spatial weights enables this research to explore the association between preschool's environment and children's play behavior in a spatial context, while spatial autocorrelation with social weights enables this research to explore the association between children's preference on school activities and preschool children's social behavior by examining their correlations in a social network context. This combined perspective provides a better understanding of the development of preschool children's social behavior than previous approaches.

The proposed methodological framework integrates spatial analysis, data mining, machine learning, spatial optimization and geovisualization techniques to discover useful knowledge and patterns from three types of experimental human activity space-time data, and can be easily extend to other types of spatiotemporal data to benefit other research fields.

The intelligent travel trip plan system has potential broader impacts for tourism (e.g. to make customized travel plans for personal guide services), and location-based services (e.g. to provide real-time touring services on GPS-enabled mobile devices). The trajectory classification model can benefit location-based services (e.g. to deliver different services to different moving objects, such as traffic/gas to drivers, landmark/shopping information to pedestrians), trajectory-based video surveillance systems (e.g. to detect abnormal movement when monitoring traffic, crowds, pedestrians etc.), and robotics (e.g. to detect and identify unknown moving objects for collision free path planning). ESDA with spatial and social weights can be applied in the social sciences for studying human social behavior and human socialization using other types of spatiotemporal data (e.g. to study Internet social behavior using micro-social data in social media websites, such as Facebook or Twitter).

Meanwhile, the proposed methodological framework contains several remaining challenges for future work. One challenge is scale (see also the Modifiable Areal Unit Problem): selecting an appropriate geographic, social or temporal scale is difficult due to the characteristics of spatiotemporal data. There are many scale options and these choices directly impact the final results. For example, for the travel trip planning problem, travelers who want to visit a city would have a detailed travel itinerary to visit attractions within the city while travelers who plan to visit a country would have a different travel itinerary to visit the most famous landmarks in this country. For studying socio-spatial pattern of preschool children's behavior, choosing semesters as a time scale will involve more children (considering that the number of preschool children in a class often changes) and more observations than using months as the time scale. It could also lead to different social weights, which will impact the results of the spatial autocorrelation tests.

Another challenge is performance: dealing with very large and complicated spatiotemporal data needs more efficient and scalable algorithms that run fast and accurately. For example, the case study in the first essay uses 118,736 geo-tagged photos and the heuristic solution for making a tourist trip plan is tested based on 2,136 discovered POIs. However, it is not feasible to directly apply the algorithms on a global scale since the overall data contain about 36 millions geo-tagged photos that are much larger than the case study data. In the second essay, the 85.4% accuracy of classification model relates to only four predefined classes and to a relative small dataset. Other movement types, such as children walking, riding a motorcycle etc., should be included to assess the performance of this model. Finally, it would also be worth testing the performance of classification on large-scale data, since only 400 selected trajectories were used in this research.