2018

Multi-Scale Flow Mapping And Spatiotemporal Analysis Of Origin-Destination Mobility Data

Xi Zhu
University of South Carolina

MULTI-SCALE FLOW MAPPING AND SPATIOTEMPORAL ANALYSIS OF ORIGIN-DESTINATION MOBILITY DATA

by

Xi Zhu

Bachelor of Science
Wuhan University of Science and Technology, 2008

Master of Science
Huazhong University of Science and Technology, 2011

Submitted in Partial Fulfillment of the Requirements

For the Degree of Doctor of Philosophy in

Geography

College of Arts and Sciences

University of South Carolina

2018

Accepted by

Diansheng Guo, Major Professor

Cuizhen Wang, Committee Member

Michael E. Hodgson, Committee Member

David Hitchcock, Committee Member

Cheryl L. Addy, Vice Provost and Dean of the Graduate School

ACKNOWLEDGEMENT

I would like to thank my advisor Dr. Diansheng Guo for his great support and help through the process of conducting this research and I am sincerely grateful for his excellent mentoring during my Ph.D. program. I would also like to thank my committee members: Dr. Cuizhen Wang, Dr. Michael Hodgson, and Dr. David Hitchcock for their valuable suggestions and guidance for this dissertation.

This research has been supported in part by the National Science Foundation under Grant No. 0748813.

I would also like to express my thanks for the colleagues in the Department of Geography, University of South Carolina. I appreciate their help and supports. Finally, I would like to thank the supports from my family.

ABSTRACT

Data on spatial mobility have become increasingly available with the wide use of location-aware technologies such as GPS and smart phones. The analysis of movements is involved in a wide range of domains such as demography, migration, public health, urban study, transportation and biology.

A movement data set consists of a set of moving objects, each having a sequence of sampled locations as the object moves across space. The locations (points) in different trajectories are usually sampled independently and trajectory data can become very big such as billions of geotagged tweets, mobile phone records, floating vehicles, millions of migrants, etc. Movement data can be analyzed to extract a variety of information such as point of interest or hot spots, flow patterns, community structure, and spatial interaction models. However, it remains a challenging problem to analyze and map large mobility data and understand its embedded complex patterns due to the massive connections, complex patterns and constrained map space to display.

My research focuses on the development of scalable and effective computational and visualization approaches to help derive insights from big geographic mobility data, including both origin-destination (OD) data and trajectory data. Specifically, my research contribution has two components: (1) flow clustering and flow mapping of massive flow data, with applications in mapping billions of taxi trips (Chapter 2 and Chapter 3); and (2) time series analysis of mobility, with applications in urban event detection (Chapter 4).

Flow map is the most common approach for visualizing spatial mobility data. However, a flow map quickly becomes illegible as the data size increases due to the massive intersections and overlapping flows in the limited map space. It remains a challenging research problem to construct flow maps for big mobility data, which demands new approaches for flow pattern extraction and cartographic generalization. I have developed new cartographic generalization approaches to flow mapping, which extract high-level flow patterns from big data through hierarchical flow clustering, kernel-based flow smoothing, and flow abstraction. My approaches represent a significant breakthrough that enables effective flow mapping of big data to discover complex patterns at multiple scales and present a holistic view of high-level information.

The second area of my research focuses on the time series analysis of urban mobility data, such as taxi trips and geo-social media check-ins, to facilitate scientific understanding of urban dynamics and environments. I have developed new approaches to construct location-based time series from mobility data and decompose each mobility time series into three components, i.e. long-term trend, seasonal periodicity pattern and anomalies, from which urban events, land use types, and changes can be inferred. Specifically, I developed time series decomposition method for urban event detection, where an event is defined as a time series anomaly deviating significantly from its regular trend and periodicity.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

Data on spatial mobility have become increasingly available with the wide use of location-aware technologies such as GPS and smart phones. The analyses of movements is involved in a wide range of domains such as demography, migration, public health, urban study, transportation and biology. A movement data set consists of a set of moving objects, each having a sequence of sampled locations as the object moves across space. The locations (points) in different trajectories are usually sampled independently and trajectory data can become very big such as billions of geotagged tweets, mobile phone records, floating vehicles, millions of migrants, etc. Movement data can be analyzed to extract a variety of information such as point of interest or hot spots, flow patterns, community structure, and spatial interaction models. However, it remains a challenging problem to analyze and map large mobility data and understand its embedded complex patterns due to the massive connections, complex patterns and constrained map space to display.

In this dissertation research, I focus on the analysis, mapping and visualization of origin-destination flow data—a specific type of spatial mobility data that concerns the movements between origins and destinations, such as human migration, taxi trips commuting, and commodity flows, among others. The goal of this research is to develop and evaluate a series of new computational and visual methods to effectively analyze and understand large origin-destination flow data. Specifically, these approaches will address a number of major challenges and research problems in analyzing big data of spatial flows,

including flow clustering, flow mapping, spatiotemporal analysis, location measure, and application development.

The dissertation work can be separated into two broad topics (1) flow analysis and mapping and (2) location analysis with mobility data. The former involves three research papers on flow clustering (Chapter 2), flow smoothing and mapping (Chapter 3). The latter topic related to the urban event detection (Chapter 4), using big data of taxi trips.

The dissertation research has makes significant contributions in methodologies for the analysis, mapping and applications of big spatial mobility data, which have become increasingly critical in understanding complex spatial and social systems such as human activities, urban structure, transportation, migration, and many others.

In the following section 1.1 and 1.2, I will introduce related work about this research.

1.1 FLOW MAPPING AND ANALYSIS

Flow mapping has long been used in a wide range of applications such as human migration, transportation, commodity flow, and commuting (Tobler 1987, Tobler 1981). However, there are two major challenges for flow mapping in displaying large data: (1) the visual cluttering problem—maps become illegible with too many flows plotted on top of each other; and (2) the modifiable area unit (MAUP) and unit size problem—flow volumes are highly correlated with unit sizes, e.g., population.

To address the visual cluttering problem, a number of approaches have been proposed in the literature based on: location aggregation, surface generation, or edge rerouting. Applications and reviews of *aggregation-based methods* for movement data can be found in (Andrienko and Andrienko 2002, Scheepens et al. 2011a, Andrienko and

Andrienko 2011), where most techniques use arbitrary area units, such as administrative boundaries, to aggregate movement data. Ferreira et al. (Ferreira et al. 2013) proposed a visual model to query origin-destination data based on user-chosen regions to aggregate flows. The drawback of this type of approaches is that aggregation can cause a significant loss of information, may omit flow patterns at local scales, and also suffer from the modifiable areal unit problem. *Surface generation approaches* produce a vector flow surface that only maps flows between geographically adjacent places(Tobler 1987), where a long-range flow is decomposed to a sequence of short flows. The limitation is that the origin and destination information for each particular flow is lost. The third group of approaches focus on minimizing edge crossing (and thus reduce cluttering) in flow maps through *edge rerouting* (Phan et al. 2005) or *edge bundling* (Verbeek, Buchin and Speckmann 2011, Holten and van Wijk 2009), which reroute or bundle edges to improve the visual clarity of flow maps. These methods are effective in producing an aesthetic representation of flow data for relatively small data sets. However, their main limitations include: (1) bundled edges make it difficult to perceive the actual connection between specific pairs of origin and destination; (2) ignoring the modifiable area unit problem and treat each flow line equally regardless of their flow volume or significance in relation to background population.

There are also a variety of methods for flow visualization based on non-spatial views, such as ordered matrices, combinations of maps and matrices, interactive OD maps(Wood, Dykes and Slingsby 2010), and exploratory visualization. Normally, interactive visualization systems do not intend to summarize the entire data set in a single flow map. Instead, they provide a non-spatial view (such as a matrix) and rely on user

interactions to select data to map and explore flow patterns through an iterative process. These interactive approaches to a certain degree avoid the visual cluttering problem but they cannot provide a clear overview of flow patterns. The matrix view approach is to visualize the origin-destination matrix rather than plotting the O-D flow data as vectors(Wood et al. 2010). In the origin-destination matrix, the rows represent the locations of flow origins as the columns represent the locations of destinations. Besides, reordering and aggregation techniques(Guo and Gahegan 2006) applied to enhance the utility of OD matrices to cope with large dataset. The limitation of O-D matrix cannot perceive the actuals routes without the geographic context. To overcomes this limitation, Wood proposed OD maps (Wood et al. 2010) that attempt to retain the geographic context as much as possible while aggregating OD flow data with regular grids. However, the choice of appropriate grid size has a significant impact on the visualization.

There are also methods for summarizing flow properties for each location using graph measures such as net migration ratio, centrality, and flow density (which is the number of flows passing a pixel)(Rae 2009). The kernel based smoothing and density estimation methods introduced earlier can be extended or customized for mapping locational characteristics of flows (but not the actual flows)(Scheepens et al. 2012). Rae (Rae 2009) generalize the O-D flow data to a flow density surface, which is a raster map view and the color of each pixel represent the number of flow passing that location. The limitation is the density of flow lines does not necessarily indicate the density of origin and destination locations. Similar kernel based smoothing and density estimation method applied on trajectory data in (Scheepens et al. 2011a, Scheepens et al. 2012, Scheepens et al. 2011b, Willems et al. 2013).

In this dissertation research, I develop and evaluate new approaches to mapping big data of spatial flows. The main idea is to first extract inherent flow patterns from big data through data mining approaches including hierarchical clustering, kernel smoothing, and generalization and then map discovered patterns with abstract flow maps that supports interactive and multi-scale exploration. Comparing to existing methods, these new approaches can cope with very large volume of data, discover unknown complex patterns, and present high-level information with new types of flow maps. These new methods are presented in Chapter 2 and Chapter 3.

1.2 EVENT DETECTION WITH MOBILITY DATA

Event detection, also referred to as anomaly detection, from spatial-temporal data has been studied extensively with traditional spatiotemporal data, such as extreme precipitation events (Wu, Liu and Chawla 2010) and outliers in meteorological data (Lu and Liang 2004). A comprehensive review can be found in (Gupta et al. 2014). Event detection from large spatial-temporal data considers both temporal and spatial information to define spatiotemporal outliers or events, whose behavioral/thematic (non-spatial and non-temporal) characteristics are significantly different from general trends in its spatial and temporal neighborhoods. Most of existing event detection techniques are based on density-based clustering method (Kut and Birant 2006, Breunig et al. 2000) or scan statistics (Kulldorff 1997).

Different from detecting events directly from spatiotemporal data points, another type of approaches aggregates data points into location-specific time series and then detects outliers from each time series. As such, the problem is converted to anomaly detection in time series. Some of these methods are based on prediction models, in which an outlier is

defined as a significant deviation from its predicted value. Each time series is individually modeled as an univariate autoregressive moving average (ARMA) process, with which outliers can be detected (Pincombe 2005, Bianco et al. 2001). However, the ARMA model is difficult to configure and often suffer from the over-fitting problem. There are extensive research on the time series analysis with satellites images, in which time series decomposition approaches are used for detecting changes within time series (Verbesselt et al. 2010a, Verbesselt et al. 2010b).

Recently, big spatial mobility data, such as geotagged social media, mobile phone usage, and taxi trips, have become increasingly available and offer unprecedented opportunities to understand the geographic and social dynamics (Liu et al. 2015). Social media check-ins, often with real-time data feed, has been widely used in event detection (Sakaki, Okazaki and Matsuo 2010, Sakaki, Okazaki and Matsuo 2013, Chae et al. 2012, Dong et al. 2015a). Recently, there has been an increasing interest in exploring both the temporal and spatial dimensions in social media data to extract and understand events at different spatial-temporal scales (Rattenbury, Good and Naaman 2007, Chen and Roy 2009, Dong et al. 2015a). Mobile phone data have been widely used in urban activity monitoring (Ratti et al. 2006, Calabrese et al. 2010), event detection (Candia et al. 2008, Traag et al. 2011, Dong et al. 2015b), population density mapping (Deville et al. 2014), and tourism management (Ahas et al. 2007, Ahas et al. 2010). A more complete review on mobile phone data analysis can be found in (Calabrese, Blondel and Ferrari 2014).

Mobile phone and social media data have the advantage of high penetration and demographic coverage. However, social media and mobile phone data have uneven and often low spatiotemporal resolutions. For example, the Call Detail Record (CDR) data, a

type of mobile phone data, records the user location when a call or text message is made or received, and the location recorded is the location of the cell tower. Therefore, the spatial resolution of mobile phone data are not very high and vary with the distribution of cell towers, ranging from hundreds of meters to kilometers. To improve the usefulness of mobile phone data, a number of approaches have been developed that either use probabilistic location inference models to enhance location accuracy (Traag et al. 2011) or combine multiple data sources to assist application-specific analyses (Calabrese et al. 2011, Liu et al. 2015, Sagl et al. 2012).

Floating car data, such as taxi trips in urban areas, have high spatiotemporal resolution and are suitable for extracting urban events with high accuracy (Calabrese et al. 2010, Zhang et al. 2015). Taxi trip data can be grouped into two kinds: trajectory data (with actual driving routes) and OD trips (with only the origin and destination of each ride). Scholz and Lu (Scholz and Lu 2014) present a method to analyze activity hot spots of urban activities with massive trajectory data. Their method defines an activity hot spot as a location with an extremely large number of activity instances during a certain hour and assumes that the theoretical distribution of activity instances across the study area and through the study time is completely random. As such, the method does not take into account of either temporal trends or periodicities in defining events or hot spots. Zhang (Zhang et al. 2015) introduce an event detection method that can consider temporal periodicity (i.e., fluctuation patterns repeated in time), which uses the Discrete Fourier Transformation (DFT) to find the length of periodicity and then define events as deviations from the periodicity. This approach does not consider the long-term temporal trend of activities, particularly for very long time series. For example, taxi pickups or drop-offs at

a specific location may gradually (or quickly) increases or decreases if the land-use type of the location changes, which should also be considered in event detection other than periodicity.

In this dissertation research, I use both long-term trends and seasonal periodicities to define events. I use the STL method (Cleveland et al. 1990) to decompose time series into three components (long-term trend, seasonal periodicity, and the remainder) and then extract events from the remainder component. In Chapter 4 I present a new approach to detecting urban events based on location-specific time series decomposition and outlier detection.

CHAPTER 2 MAPPING LARGE SPATIAL FLOW DATA WITH HIERARCHICAL CLUSTERING

2.1 ABSTRACT

It is challenging to map large spatial flow data due to the problem of occlusion and cluttered display, where hundreds of thousands of flows overlap and intersect each other. Existing flow mapping approaches often aggregate flows using predetermined high-level geographic units (e.g. states) or bundling partial flow lines that are close in space, both of which cause a significant loss or distortion of information and may miss major patterns.

In this research, I developed a flow clustering method that extracts clusters of similar flows to avoid the cluttering problem, reveal abstracted flow patterns, and meanwhile preserves data resolution as much as possible. Specifically, our method extends the traditional hierarchical clustering method to aggregate and map large flow data. The new method considers both origins and destinations in determining the similarity of two flows, which ensures that a flow cluster represents flows from similar origins to similar destinations and thus minimizes information loss during aggregation. With the spatial index and search algorithm, the new method is scalable to large flow data sets. As a hierarchical method, it generalizes flows to different hierarchical levels and has the potential to support multi-resolution flow mapping. Different distance definitions can be incorporated to adapt to uneven spatial distribution of flows and detect flow clusters of

different densities. To assess the quality and fidelity of flow clusters and flow maps, we carry out a case study to analyze a data set of 243,850 taxi trips within an urban area.

2.2 INTRODUCTION

Large data on geographic mobility such as migration, commuting, movements of goods and disease spread have become increasingly available due to the wide adoption of location-aware technologies. In our research, we focus on the origin-destination flow data, a specific type of geographic mobility data that concerns the origin and destination of each movement but ignores the actual trajectory route, for example, taxi trip data that has origin and destination points for each passenger ride. This form of data can also be referred to as flow data or spatial interaction data. It is a challenging problem to map and understand patterns in massive origin-destination data due to the problem of occlusion and cluttered display, where thousands or millions of flows overlap and intersect each other.

Location-based graph measures and summary statistics, such as in-flow, out-flow, and net- flow ratios, are often used to understand location characteristics and spatial patterns of mobility data (Guo et al. 2012). Such measures project the data to a certain perspective but do not allow direct understanding of the connections among locations. Flow maps, on the other hand, can directly plot links among locations and show patterns of geographical movements (Tobler 1987, Tobler 1981). However, traditional flow maps are not capable of mapping large flow data. A typical origin-destination flow data set, such as migration paths among US counties or taxi trips in a city, can easily have millions of origin-destination pairs. A number of new flow mapping approaches have been proposed to visualize and discover patterns in large flow data sets (Phan et al. 2005, Cui et al. 2008,

Guo 2009, Holten and van Wijk 2009, Andrienko and Andrienko 2011, Verbeek et al. 2011).

This article presents a new flow-mapping approach based on flow clustering to effectively generalize large point-to-point spatial flow data, discover major flow patterns, and preserve data resolution as much as possible within the map space. The approach can process large origin-destination flows and adapt to skewed spatial distributions, i.e. flow clusters of different spatial densities. Unlike existing approaches that aggregate locations or bundle flow lines, we directly cluster flows based on both origin and destination similarities among flows. The key contribution of the approach is a new agglomerative clustering method specifically designed for origin-destination flows and scalable to large data size. We demonstrate and evaluate the usefulness of the new method with a case study on taxi trip analysis.

2.3 METHODOLOGY

In this section, we present our new flow clustering and mapping method that can extract flow clusters and render a visually clear flow map to generalize patterns in massive spatial flows. First, the flow clustering method considers both origins and destinations in determining the similarity of flows, ensures that a flow cluster represents flows from similar origins to similar destinations. Second, with spatial index and search algorithm, the method is scalable to large data sets. Third, it is a hierarchical clustering method and thus has the potential to support multi-resolution flow mapping. Different distance definitions can be incorporated in the method. In this article we focus on a shared-nearest-neighbor distance measure to capture flow clusters of different densities.

2.3.1 Overview

Let $T = \{T_i\}$ be a set of origin-destination flows, where $T_i = \{O_i, t_{oi}, D_i, t_{di}\}$ is a directed flow that starts at origin location O_i and time t_{oi}, and ends at destination D_i and time t_{di}; $n = |T|$ is the number of flows. The time stamp for flows is optional, which can be omitted if all flows are in the same time period. Let $O = \{O_i = <ID_i, X_i , Y_i> \}$ be all origin locations and $D = \{D_j = <ID_i, X_i , Y_i> \}$ be all destination locations that are involved in T. Each location has a pair of spatial coordinates $<X_i, Y_i>$ and a unique ID. We treat O and D separately in the method as origins and destinations have different meanings for flows. Conceptually, our agglomerative flow clustering method consists of the following steps:

1. Build a spatial index based on a contiguity definition to allow efficient retrieval of the nearest neighbors for a given origin/destination and the nearest flows for a given flow;
2. Define a distance (or dissimilar) measure between flows, which should allow the detection of flow clusters of different spatial densities;
3. Cluster flows with an efficient agglomerative clustering procedure, which iteratively groups flows into a hierarchy and stops at a level controlled by given parameters;
4. Render a flow map with the discovered flow clusters $C = \{C_l\}$, which is a generalization of the original set of flows $T = \{T_i\}$, $|C| \ll |T|$.

2.3.2 Neighborhood Search for Points and Flows

To facilitate presentation and discussion of the spatial index and subsequent clustering steps, the following definitions on neighborhoods for points and flows are needed:

Definition 1.1 (Euclidean Neighborhood of a point) The Euclidean neighborhood of an origin point O_p is defined as $EN(O_p, r) = \{O_q \in O \mid EuclideanDist(O_p, O_q) <= r\}$, where r is a constant search radius. Similarly, the definition applies for destination points in D.

Definition 1.2 (K-Nearest-Neighbor (KNN) Neighborhood of a point) The k-nearest-neighbor neighborhood of an origin point O_p is defined as $KNN(O_p, k) = \{O_q \in O | O_q \in EN(O_p, r_p) \text{ and } |EN(O_p, r_p)| = k)\}$, where k is the number of nearest neighbors and rp is the search radius that is specific for Op and determined by k. Similarly, the definition applies for destination points in D.

Definition 2 (KNN Neighborhood of a flow) The KNN neighborhood of a spatial flow Tp is $FN(T_p, k) = \{T_q \in T | O_q \in KNN(O_p, k) \text{ and } D_q \in KNN(D_q, k)\}$, where O_p, O_q are the origins and D_p, D_q are the destinations of flow T_p and T_q.

In Figure 2.1, the two circles in the left map represent the Euclidean neighborhood of the origin (blue) and destination (green) of a flow (red). The two circles in the right map show the K-Nearest-Neighbor neighborhoods (k = 10) for the origin and destination of the flow q at the center. Flows p1 and p2 are contiguous to flow q with the KNN neighborhood definition (right map) but not for the Euclidean neighborhood definition (left map).

We first build spatial R tree indices on the origin and destination points, separately, with which we can quickly find the nearest neighbors for each origin and each destination. Each origin (or destination) has a stored list of flows that involve the origin (or destination).

In our case study of taxi trips, each origin (or destination) is unique, i.e. belongs to a unique flow. For other data sets it is possible that one origin (or destination) may be involved in multiple flows, e.g. a county has migration connections to many other counties.

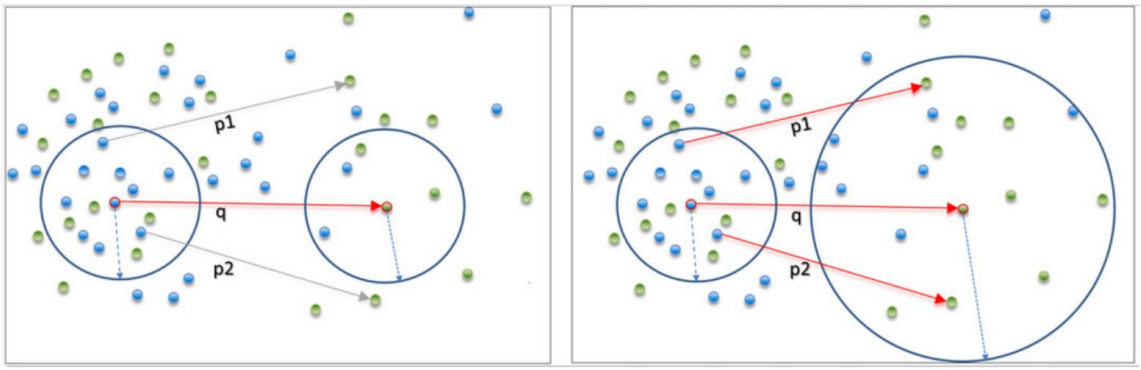

Figure 2.1 An illustration of contiguity definitions for origins, destinations, and flows. Flows p1 and p2 are contiguous to flow q with the KNN neighborhood definition (right map) but not for the Euclidean neighborhood definition (left map)

Given a flow T_p (with origin O_p and destination D_p) and a value of k, we find the k nearest origins $KNN(O_p, k) = \{O_q\}$ and k nearest destinations $KNN(D_p, k) = \{D_q\}$; then find the neighboring flows $FN(T_p, k) = \{T_q\}$ by taking the intersection of the flow sets that involve $\{O_q\}$ and $\{D_q\}$. A contiguous flow pair will be created between T_p and each of its neighboring flows in $\{T_q\}$. The subsequent clustering will only use these contiguity pairs to derive clusters. The larger the k value is, the more contiguity pairs will be created, and this demands more computational time in the later clustering step. On the other hand, k should be sufficiently large to ensure that the contiguity graph connects most flows. In Section 2.5 we will explain how to configure the k value.

2.3.3 Shared-Nearest-Neighbor Flow Distance

The distance or dissimilarity measure for two flows p and q, as defined in Equation (1), is based on the number of shared nearest neighbors (SNN) between their origins and destinations, respectively. This similarity extends the SNN distance measure used for spatial point clustering (Jarvis and Patrick 1973, Guo et al. 2012), which can find natural groupings of different point densities. Figure 2.2 shows an illustrative scenario for the distance calculation (with k = 7). The circle centered on the origin point of flow p covers seven nearest origins, i.e. KNN(O_p, 7). Similarly, the other three circles represent KNN(O_q, 7), KNN(D_p,7) and KNN(D_q,7), respectively. Note that origins and destinations are treated separately. Since |KNN(O_p, 7)∩KNN(O_q, 7)| = 2 and |KNN(D_p, 7)∩KNN(D_q, 7)| = 3, the distance between flows p and q is 1- (2/7*3/7) ≈ 0.87. If the two flows do not share any origin or destination neighbors, the distance is 1; conversely, if their origin and destination neighborhoods are both identical, the distance is zero.

$$dist(p,q) = 1 - \frac{|KNN(Op,k) \cap KNN(Oq,k)|}{k} * \frac{|KNN(Dp,k) \cap KNN(Dq,k)|}{k} \qquad \text{Equation 1}$$

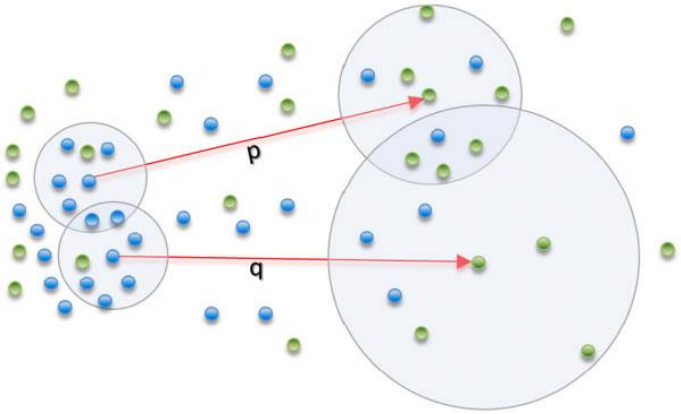

Figure 2.2 An illustration of the SNN distance measure between two flows, with k = 7. Blue points are flow origins and green points represent destinations

2.3.4 Agglomerative Flow Clustering Algorithm

Our clustering algorithm iteratively merges flows to form a hierarchy of flow clusters. Below is the algorithm outline, requiring only one input parameter k – the number of origin/destination neighbors. Note that, although we use the shared-nearest-neighbor distance to discover flow clusters of different densities, other distance measures (such as the Euclidean distance) can be easily integrated in this algorithm to suite specific needs.

Algorithm 2.1: Agglomerative Flow Clustering

Input: $T = \{T_i \mid 1 \leq i \leq n\}$ – a set of origin-destination flows; and k – the number of nearest neighbors used in calculating distance.

Output: A set of flow clusters $C = \{C_l \mid 1 < l << n\}$

Steps:

1 Identify neighboring flows for each flow with a search radius k and create contiguous flow pairs, as explained in step 1.

2 Calculate the distance for each contiguity flow pair according to Equation 1, as explained in step 2;

3 Sort all contiguous flow pairs to an ascending order based on their distances;

4 Initialize a set of flow clusters by making each flow a unique cluster, i.e. $C = \{C_l\}$ and $C_l = \{T_l\}$, $1 \leq l \leq n$; and

5 For each contiguity flow pair (p, q), following the above ascending order:

   a. Find the two clusters C_x and C_y that p and q belong to: $p \in C_x$ and $q \in C_y$;

   b. Calculate the distance $dist(C_x, C_y)$ between C_x and C_y (see text below for detail); and

   c. If $x \neq y$ and $dist(C_x, C_y) < 1$, merge them: $C_x = C_x \cup C_y$ and $C = C \backslash C_y$

2.3.5 Algorithm Configuration and Complexity

The k parameter has two main effects. First, it determines the contiguity and the similarity measure among flows. We only merge flows (or clusters) that are contiguous. If k is too small, the contiguity graph for flows will have many disconnected components and tend to generate many small clusters. Second, a large k value will create many more contiguity pairs and thus demands more computational time in the later clustering step. We designed a simple but effective heuristic technique to set the parameter k, inspired by the parameter selection in DBSCAN (Ester et al. 1996).

We define m-dist for a flow *p* as the minimum k value (i.e. number of nearest neighbors for point neighborhood) in order to find m neighboring flows for p, according to Definition 2. For a given m, we calculate the m-dist value (i.e. k) for each flow and sort all m-dist values to render a plot. For the case study of taxi trips, its m-dist plot in Figure 2.3 shows that, with k = 1,000, more than 95% of original flows can find at least one neighboring flow, and about 70% of flows can find seven or more neighboring flows. This is sufficient for the purpose of ensuring reasonable spatial contiguity. For example, in a polygon data set or a Delaunay triangulation of point data, each object on average has 4–5 spatial neighbors.

The inset diagram in Figure 2.3 shows the distribution of the number of flow neighbors with the taxi data for k = 1,000, where on average each flow has 10 flow neighbors. As such, the m-dist plot can help us make informed choice of the k value. According to our experiments, k = 1,000 works very well for the taxi data. In the case study section, we provide a brief sensitivity evaluation by comparing the clusters and patterns with different k values.

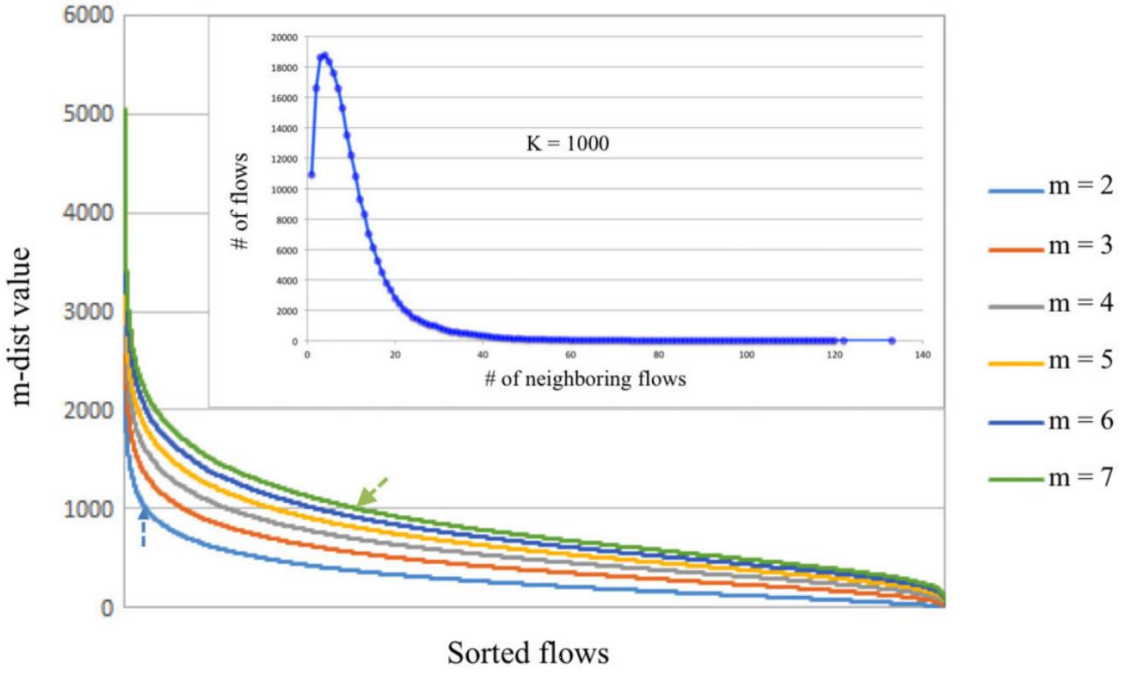

Figure 2.3 Sorted m-dist values for the taxi trip data. When k = 1,000, flows to the right of the blue arrow have one or more neighboring flows; while flows to the right of the green arrow have seven or more neighboring flows. The inset diagram shows the distribution of the number of flow neighbors with k = 1,000, where on average each flow has 10 flow neighbors

The computational complexity of the method involves three steps: (1) Building a contiguity index for flows, which needs to find k nearest points for both the origin and destination of each flow, and then find flows with both origin and destinations located within the neighborhood of the flow. With the assistance of a spatial index such as R-tree, the complexity of this step is O(knlogn); (2) Calculating and sorting the distances of contiguity pairs. There are at most O(mn) contiguity pairs, where m is the average number of flow neighbors. For each pair of flows, it takes O(k) time (with the help of Hash mapping) to count shared neighbors and calculate its distance according to Equation 1, and O(mnlog(mn)) time to sort all pairs. The overall complexity for this step is O(kmnlog(mn));

(3) The complexity for the clustering step is O(mnk). Therefore, the overall complexity of Algrithm 1 is O(kmnlog(mn)), m << k << n.

2.3.6 Mapping Flow Clusters

Given a set of flow clusters derived with previous steps, we calculate a mean flow for each cluster using the centroid of origins, centroid of destinations, and the total flow volume of flows in the cluster. It is straightforward to render a flow map with the clustered flows. To verify the flow patterns discovered by our method, we calculate locational measures, such as in-flow, out-flow, net-flow ratios for each local neighborhood, and compare their spatial patterns with the flow clusters. One can conduct various analyses with the new flow clustering and mapping method. In this article we will elaborate on one possible analysis, i.e. analyzing the spatiotemporal patterns of mobility, based on the flow clusters of different time periods. We present a case study on taxi trip analysis in the following section.

2.4 CASE STUDY

We applied the developed flow clustering and mapping method to analyze a dataset of 243,850 taxi trips for a working week (five days) in the downtown area of Shenzhen, China. Each taxi trip has an origin GPS point, a destination GPS point, and time (when the passenger was dropped off). Without losing generality, we can assume that each GPS point in the dataset is unique, collected independently with about 10 m accuracy. If there were identical points, we still treat them as different points. Therefore, there are 243,850 origins and the same number of destination points. The spatial distribution of these points are

highly skewed, with some areas having about 30,000 points per square kilometer while other areas have only 500 points or less per square kilometer. The point distribution also follows the road network. The average trip distance is 5.64 km, and taxi trips often travel between areas with different point densities. It is difficult to map and understand such massive flow data in its original form. Figure 2.4 shows 1% of the flows (left map) and 10% of the flows (right map), both of which cannot offer many insights on the flow patterns in the data.

2.4.1 Taxi Trip Clustering and Mapping

Based on the m-dist plot in Figure 2.3, we set k = 1,000, with which most of the flows can find one or more neighbor flows, with an average of 10 neighbors for each flow. There are 11,000 flows (about 4%) which cannot find any neighboring flows; each will form a cluster of its own (and likely be excluded in the final mapping). A pair of flows or flow clusters can only be merged if their distance is less than 1, meaning that they must share at least one point in their origin neighborhoods and one point in their destination neighborhoods (see Equation 1).

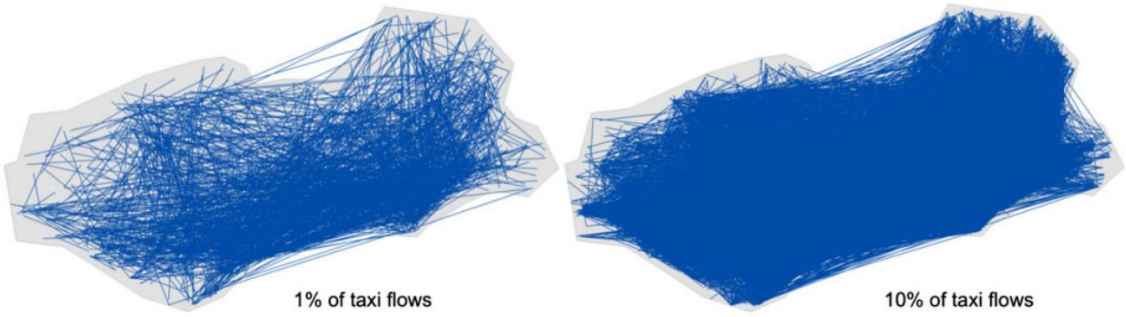

Figure 2.4 Flow maps of random samples of taxi trips. The left map shows 2,500 flows (i.e. 1% of the original data) and the right map shows 25,000 flows (10%)

The original 243,850 flows are grouped into 18,969 flow clusters, in which the largest cluster has 732 flows. There are 347 clusters with size > 100 (i.e. having more than 100 flows in the cluster) These 347 clusters together contain 51,924 original flows (>21%). There are 1,309 clusters with a size > 50, which together contain 118,022 flows (about 50% of the original data). There are 5,269 clusters with at least 10 flows, covering nearly 90% of the original flows. Therefore, the clustering method generalizes the large input flow data into a relatively small set of clusters, which is only about 2% of the original data size and yet covers 90% of the original data. The flow map in Figure 2.5 shows the top flow clusters, where each curved line represents a cluster, starting from the centroid of its origin to the destination centroid, with the arrow pointing to the destination. Colors and line widths represent cluster sizes. Note that two flows or clusters must share both an origin point and a destination point in their neighborhoods, respectively. Therefore, each flow cluster is spatially compact (a relative term that depends on the distance definition adopted in the algorithm).

We also calculate a smooth surface of net flow ratio by: (1) partitioning the area into grid cells; (2) finding k nearest origins/destination points (and thus flows) to each cell; and (3) calculating a net flow ratio (inFlow – outFlow) / (inFlow + outFlow) for each grid, which are mapped in Figure 2.4 as the background, with darker colors representing high net flow ratios (i.e. more incoming taxi trips than outgoing trips). This generalized flow map reveals major flow patterns in the data and important places/regions. For example, in the flow map we can easily recognize a number of hubs including the Huanggang Port on the border with Hong Kong, train station, subway stations, and other significant centers.

The flow map patterns not only match the location measure patterns well but also reveal more specific flow patterns with clear connection, direction, and flow strength indicators.

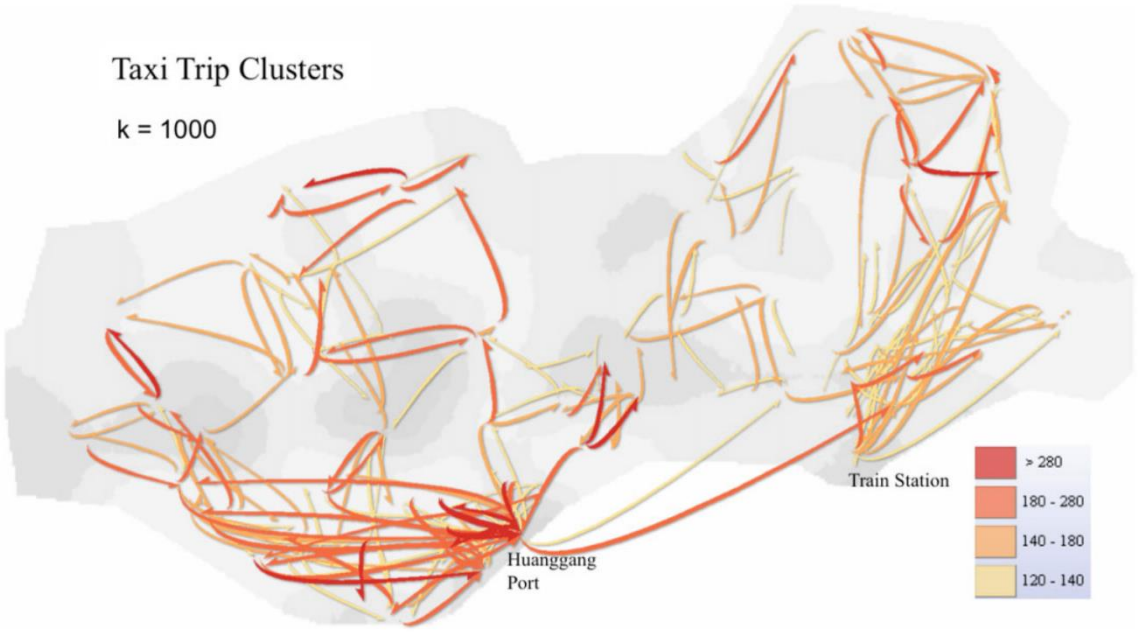

Figure 2.5 Top flow clusters of taxi trips. The background map represents the smoothed net flow ratio, with darker colors for high net flow ratios

2.4.2 Spatiotemporal Analysis of Taxi Trip Clusters

To examine temporal patterns of taxi trips, we divide a day into six four-hour time windows: 5–9 a.m., (morning), 9 a.m.–1 p.m. (noon), 1–5 p.m. (afternoon), 5–9 p.m. (evening), 9 p.m.–1 a.m. (night), and 1–5 a.m. (early morning). The morning and evening periods, in particular, will capture the heavy–traffic hours of a day. We calculate the frequency of flows in each time window for each of the 1,309 flow clusters that have 50 or more flows. Then we map two subsets of the clusters in separate flow maps, one showing flow clusters dominated by morning flows (Figure 2.6) and the other showing flow clusters dominated by evening flows (Figure 2.7). Figure 2.8 shows the color legend for the temporal frequency of flows in each subgroup. The net flow ratio values for the specific

time period are also mapped in Figure 2.6 and Figure 2.7 in the background. In these two time-specific flow maps, we can see clear and distinctively different flow patterns. In the morning most taxi trips are from residential areas to transportation hubs (e.g. ports, train and subway stations) or commercial/industrial areas. In the evening, the flow patterns are almost the opposite.

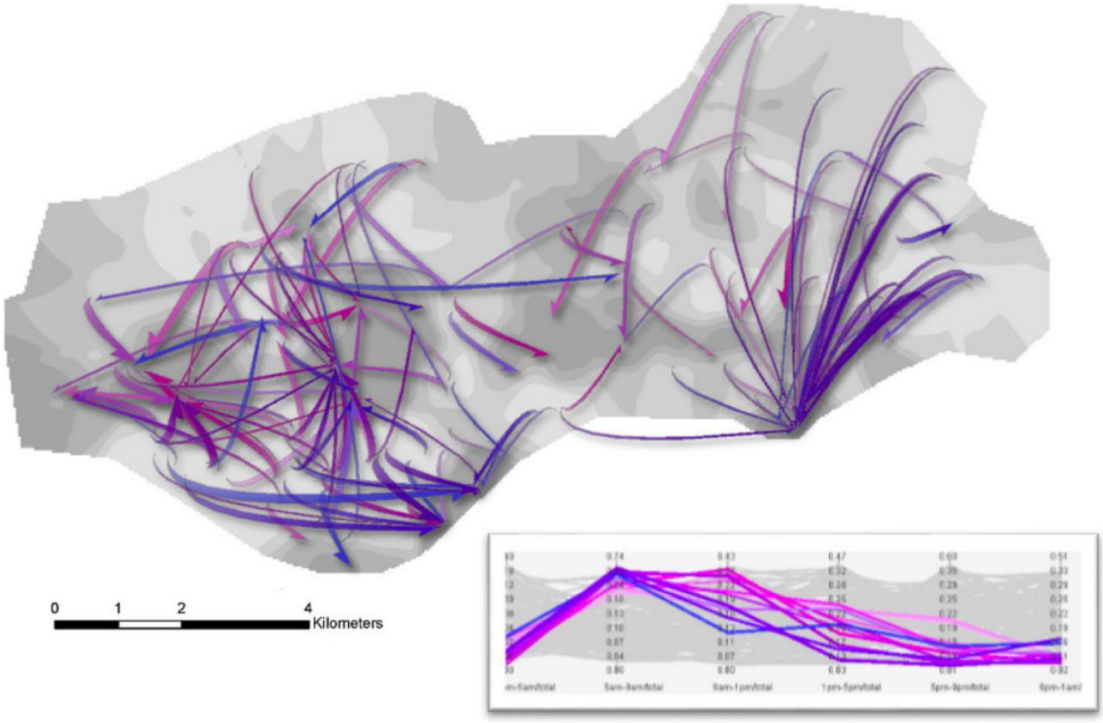

Figure 2.6 Taxi trip flow patterns during the morning traffic hours (i.e. 5 a.m. to 9 a.m.)

Note that the subset of clusters in Figures 2.5, 2.6 and 2.7 are from the same hierarchical clustering result. They are chosen based on different criteria, with Figure 2.5 showing the largest flow clusters of all time, Figure 2.6 showing clusters dominated by morning flows, and Figure 2.7 showing clusters dominated by evening flows. The clustering process not only generalizes the flow map but also enables analysis of flow characteristics and their spatial distribution.

2.4.3 Parameter Configuration and Comparison

In the above analysis, we set k = 1,000 according to the m-dist plot in Figure 2.3 and our experiments. As explained earlier, if k is too small, the contiguity graph for flows will have many disconnected components; if k is too large, it will create many more contiguity pairs and demand more computational time in the later clustering step. To examine the sensitivity of results to the k configuration, the clustering results for k = 1,500 and k = 2,000 are shown in Figure 2.9, Figure 2.10 and the result for k = 1,000 is shown in Figure 2.5. Comparing the three maps, we can see that: (1) the patterns in general are similar although the sizes of the top clusters are different; and (2) the larger the k value is, the larger the top clusters are as clusters can be merged further with larger neighborhoods (defined by k). This result shows that the clustering result is not very sensitive to k configuration and responds to different k values in a predictable way.

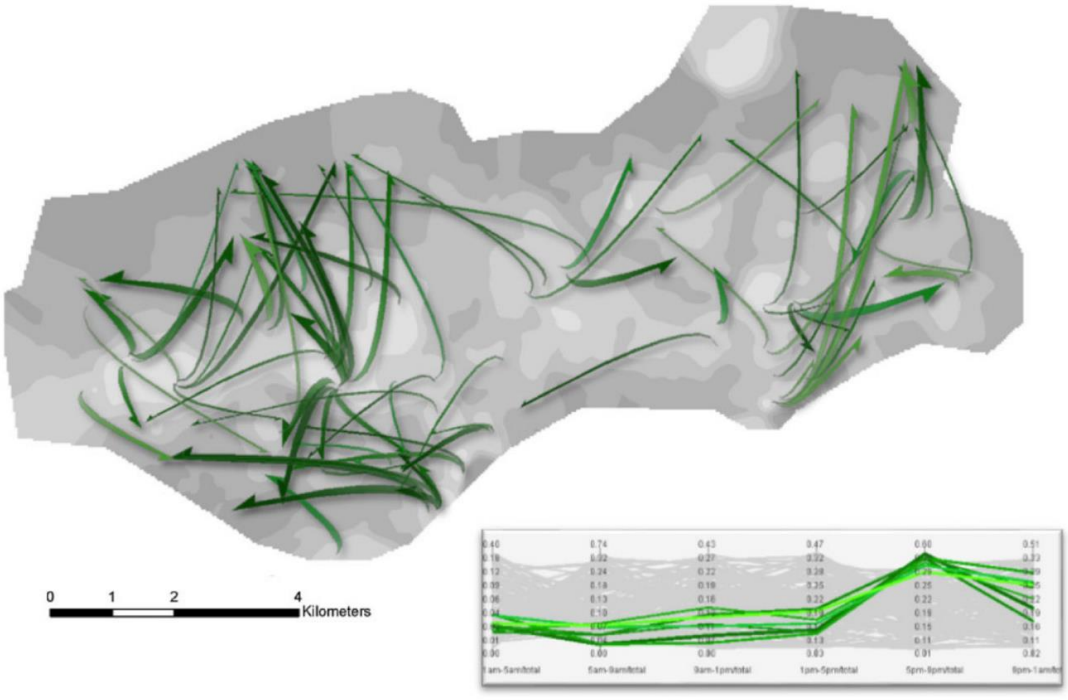

Figure 2.7 Taxi trip flow patterns during the evening traffic hours (i.e. 5 p.m. to 9 p.m.)

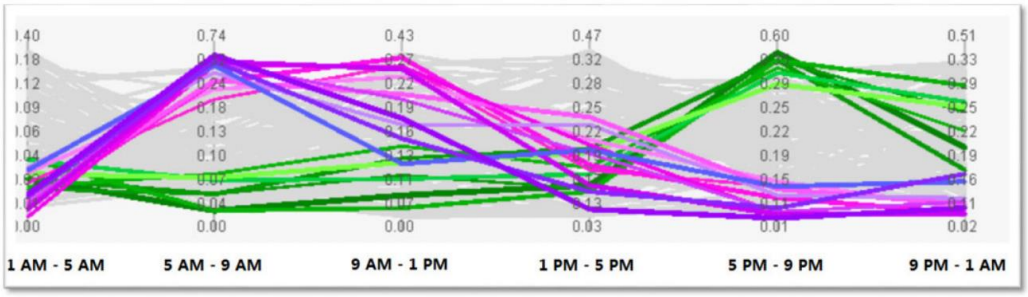

Figure 2.8 Multivariate legend of the flow colors in Figures 2.6 and 2.7

Figure 2.9 Comparison of clustering results with k = 1,000 (see Figure 2.5), k = 1,500 and k = 2,000

Figure 2.10 Comparison of clustering results with k = 1,000 (see Figure 2.5), k = 1,500 and k = 2,000

2.5 CONCLUSION

This article presents a new flow clustering and mapping method with needed capability and scalability for handling large origin-destination flow data. It can be used for exploratory analysis, visualization, or communication of flow data and patterns. The key component in the method is an agglomerative clustering method, which requires only one input parameter k, the number of nearest neighbors. If other distance measures are used, k is not needed but other parameters may be needed, related to the chosen distance measure. The configuration of k is critical and we provided an approach based on the m-dist plot to help make informed decisions. Our experiments show that the clustering result is not very sensitive to the k value.

We carried out a case study with taxi trip data in Shenzhen to evaluate the usefulness of the proposed approach. Results show that the proposed method can effectively process large datasets and discover major flow patterns, which can also support the analysis of temporal variations, avoiding the weakness in other existing methods related to arbitrary aggregation of locations. The method can incorporate different distance and contiguity definitions to address different data or application needs. It is also possible to add constraints such as a maximum Euclidean distance threshold to avoid merging flows that are not close geographically while still allowing the detection of flow clusters with different densities.

Future work is needed to better balance and automatically determine the appropriate level of flow clusters to optimize the flow map layout. This will require both intelligent algorithms and interactive human inputs. The robustness of the proposed approach needs further evaluation with both real datasets and specifically designed synthetic datasets with

various known patterns. Our current mapping approach does not allow much user interaction, which is partially due to our belief that an information-rich and static flow map is very important for communication purposes. Of course, supporting innovative user interactions will add additional benefits and will be explored in future work. Although the method is reasonably efficient, future work is needed to develop approximation extensions for analyzing even larger data sets with billions or more of points and flows. Another aspect that needs improvement is the ability to handle non-point flow data such as the US county-to-county migration.

CHAPTER 3 INTERACTIVE MULTI-SCALE FLOW MAPPING WITH KERNEL SMOOTHING MAPPING

3.1 ABSTRACT

Flow mapping of large origin-destination data has long been a challenging problem because of the conflict between massive location-to-location connections and the limited map space. Current approaches for flow mapping only work with small dataset or have to use arbitrary data aggregation, which not only cause a significant loss of information but may also produce misleading maps. In this paper, we present a multi-scale flow map generalization approach that can extract flow structures at different scales and facilitate the analysis and visualization of big spatial flow data. The approach is built on a novel flow data aggregation method, which uses flow-based kernel density estimation and a greedy search of local maximum of dense flows. Given a scale, it discovers inherent and abstract flow patterns appropriate for the scale and naturally supports interactive and multi-scale flow mapping. The pattern scale is controlled with the kernel bandwidth and the pattern resolution is determined with the search radius. To demonstrate the approach and assess its effectiveness, a case study is carried out to map billions of taxi trips within the New York City.

3.2 INTRODUCTION

Origin-Destination (OD) flows, a specific type of geographic mobility data, record the origin and destination of movements but ignore the actual trajectory route. Such data have become increasingly available and accurate due to the wide adoption of location-aware technologies, including taxi trips, county-to-county migration, cell phone calls, commuting data, and disease spreading. Mapping and understanding massive origin-destination flow data is fundamentally important for a wide range of research fields and decision makings in practice such as demography, urban planning, transportation, and epidemiology.

Geographic flow visualization and visual analytics, such as flow map (Tobler 1987, Tobler 1981) and space-time cube (Kraak 2003, Kraak and Koussoulakou 2005), are commonly used in OD flow data analysis. However, existing approaches suffer the severe problem of occlusion and cluttered display when the data is relatively large. Traditional flow map generalization usually aggregate large flow data by predefined areas, which is arbitrary and suffers from the MAUP. As stated in (Rae 2011): "flow mapping is something of a disciplinary laggard", especially in an big data era that mobility is one of the key aspects for various geographic research problems. In addition to the massive data volume, another challenge in analyzing mobility data is that mobility patterns may exist at different spatial scales (Laube and Purves 2011, Soleymani et al. 2015), where cross-scale analysis is of great importance (Fryxell et al. 2008, Nathan et al. 2008).

In this paper, we propose the approach to enable multi-scale flow mapping. The main contribution of the approach in this paper is twofold. First, we put forward a general framework for multi-scale and multi-resolution flow mapping, where the scale is related to

the kernel bandwidth of flow smoothing and the resolution is related to the search radius in flow selection. Second, we designed a new flow map generalization algorithm based on local maximal search. With these two contributions, the approach can (1) discover and present generalized movement patterns at different spatial scale and resolution; and (2) enables interactive exploration of big flow data across scales with overview and zoom-in capabilities. We carried out a case study to analyze and map billions of taxi trips in New York City.

3.3 RELATED WORKS

3.3.1 Flow map and cartographic design

Flow map has long been used in a wide range of applications such as human migration, transportation, commodity flow, and commuting(Tobler 1987, Tobler 1981). There are a few recent research works on the cartographic design principles for the symbolization and layout of flow map based on quantitative analyses of users' perception performance (Koylu and Guo 2016, Jenny et al. 2016, Jenny et al. 2017) and similar studies have also been conducted in the graph drawing community (Alper et al. 2013, Dwyer et al. 2009, Xu et al. 2012). One of the design principles is to minimize edge crossing to achieve visual clarity, for which a number of strategies have been proposed such as edge rerouting/bundling (Phan et al. 2005, Cui et al. 2008, Buchin, Speckmann and Verbeek 2011) or producing non-branching flows by adjusting the curvature of flow lines (Jenny et al. 2017). However, there are also a few design principles remain debatable. For example, Xu et al. (2012) found that user prefer straight lines over curved lines while the study in (Jenny et al. 2016, Purchase et al. 2012) shows curved lines are more effective than straight

lines. For curved flow lines, Jenny et al. (2016) suggest to use symmetric flows rather than asymmetric flows, while others recommend the use of asymmetric flow lines to encode direction (Guo 2009, Ware, Kelley and Pilar 2014).

Koylu and Guo (2016) show with experiments that the choice of symbolization type may depend on different tasks (e.g., the identification of the dominant flow directions or the identification of strongest flows) and different data patterns also have a strong effect on task performance and pattern perception in flow maps. It is difficult to generalize a universal set of design guidelines to create flow maps for different tasks and across various datasets. Interactive techniques may provide a viable solution, with which one can customize the symbolization and layout of flow maps according to the data, task, and the user's preference.

The goal of this study is to flow map generalization, which is different from these cartographic design researches. These design principles can improve visual clutter problem at certain extend, but it is only can be applied to small flow datasets (up to one hundred). It is necessary to apply data abstraction or aggregation when cope with large volume of flow data. Our proposed method focus on pattern extraction from very large and dense origin-destination flow data, and then visualize the extracted patterns. The researches mentioned above can provide design guidelines to visualize extracted flow patterns.

3.3.2 Flow map generalization

Mapping large volume of origin destination flows is a challenging research problem, since the flow map suffers from the visual cluttering problem while the data volume increased. As Andrienko et al. (2008) suggest, current visualization research need higher

degree of abstraction that can extract and visualize high-level abstract patterns from data. One type of such data reducing approaches is to aggregate OD data by using large geographic units. Researchers were looking for optimal way to define locations for aggregating movement data rather than define the regions arbitrarily. For instance, Guo (Guo 2009) proposed a spatially constrained graph partition method that can construct a hierarchical regions. Andrienko (Adrienko and Adrienko 2011) developed a computational method that uses trajectory points' coordinates to partition the territory into suitable places. The drawback of this type of approach is that aggregation can cause a significant loss of information, skip flow patterns at local scales, and suffer from the modifiable areal unit problem.

Another type of data aggregation approaches simplify the flow data using clustering or cluster detection method. For instance, Zhu (Zhu and Guo 2014) proposed a clustering method to aggregate OD data based on the spatial similarity. Tao (Tao and Thill 2016) applied hot spot detection method on flow data to detect spatial flow clusters. Guo (Guo and Zhu 2014) extract flow patterns by applying kernel density estimation on flow data. Yan (Yan and Thill 2009) adopts self-organizing maps serves as the data mining engine to reduce data complexity. However, clustering method can reduce the data size at the cost of data resolution, which is coincides with the idea that any knowledge gain may accompany by information loss in data mining process. In this research, our proposed method belongs to this category. To cope with the problem of information loss, our proposed method can detect flow patterns at multiple spatial scales. The flow map at lower scale serves as complementary view for flow map at higher scale. Besides, our proposed method implements multi-resolution flow map by interaction techniques. A visual control interface

such as a slider bar, can be used to select a balance point between generalization level and information resolution.

3.3.3 Flow data visualization and visual analytics

There are also a variety of methods for flow data visualization and visual analytics such as location based visualization, matrix view, raster map in order to bypass the visual clutter problem. Alternative strategies visualize the location measures such as net flow ratio for different time durations derived from OD flow data(Guo et al. 2012), and it provides insights of characteristic of locations. The drawback of the location based analysis is the link between locations lost. Andrienko (Andrienko et al. 2016) presents a spatial and temporal abstraction method to represent location measure by diagram maps instead of flow maps, and the proposed method visualize the flow angle and distance in the diagram map to retain the links between locations.

The matrix view approach is to visualize the origin-destination matrix rather than plotting the O-D flow data as vectors (Ghoniem, Fekete and Castagliola 2004). In the origin-destination matrix, the rows represent the locations of flow origins as the columns represent the locations of destinations. Besides, reordering and aggregation techniques (Guo and Gahegan 2006) applied to enhance the utility of OD matrices to cope with large dataset. The limitation of O-D matrix cannot perceive the actuals routes without the geographic context. To overcomes this limitation, Wood proposed OD maps (Wood et al. 2010) which attempts to retain the geographic context as much as possible.

Raster map generalized the O-D flow data to a flow density surface, which is a raster map view and the color of each pixel represent the number of flow passing that

location. The limitation is the density of flow lines does not necessarily indicate the density of origin and destination locations. Similar kernel based smoothing and density estimation method applied on trajectory data in (Scheepens et al. 2011a, Scheepens et al. 2012, Scheepens et al. 2011b). Note, the kernel based smoothing and density estimation method is different from our model in this paper.

Besides, there is a research trends to apply visual analytics on O-D flow data. The visual analytics utilize the computational power to corporate human capabilities to understand the data(Thomas and Cook 2006). Recently, filtering and selection method is applied to improve the visualization efficiency. This method downscale to a data subset according to a user-specified query. Ferreira et al. (Ferreira et al. 2013) proposed a visual model that supports spatiotemporal queries of origin-destination data, which is based on user's choices of regions to aggregate flows. (Boyandin et al. 2011) proposed a view contains two maps, and place the origin and destination separately on these two map in order to avoid line occlusion. The selection approaches to a certain degree the visual cluttering problem but the limitation is they cannot provide a clear holistic overview of spatial flow patterns. In this paper, we proposed a visual analytic framework for multi-scale flow mapping. Specifically, we provide an overview for the whole dataset in each scales, and zooming and panning operations allow the users switch between the different scales and views. Besides, the proposed method also can facilitate multi-resolution flow map exploration by providing a visual interface to enable real-time parameter adjusting.

3.4 METHODOLOGY

In this section, we present our approach to mapping massive origin-destination flow data at different spatial scales and generalization levels. The main idea is to treat a flow as

a spatial point in 4D space to estimate the density distribution of flows with a kernel density model, and then to generalize the generated density surface as several local maxima points. The extracted local maxima points in 4D space can be visualized as flow lines on 2D flow map. With different smoothing and generalization parameter settings, an interactive flow map enable us to observe the flow patterns at different spatial scales and generalization levels. This method intends to gain a highly abstraction from the original dataset without too much artificial interference.

A kernel density estimation is closely similar to a histogram, both of which are classical methods to summarize large data points, but the histogram is sensitive to the anchor point and bin size. The traditional methods aggregate the origin-destination flow data arbitrarily by predefined geographical regions or administrative boundary, which can be analog to histogram. Different from the traditional method, the advantage of kernel density estimation can preserve the characteristics of the dataset in the data abstraction process.

Figure 2.1 is an illustration of the overall methodology. Specifically, the method consists of the three steps: (1) estimating density value by considering flows within its neighborhood; (2) selecting local maxima based the result of kernel density estimation and visualize extracted local maxima from 4D space as flow lines on flow map; and (3) multi-scale interactive mapping with a set of different parameters for smoothing and generalization. Below is an illustration of our proposed method, and we described the flow density estimation (Section 3.4.1) and flow generalization (Section 3.4.2) in detail. Besides, we applied this method on a set of flow dataset, which contains a billion of taxi trips, in the case study section.

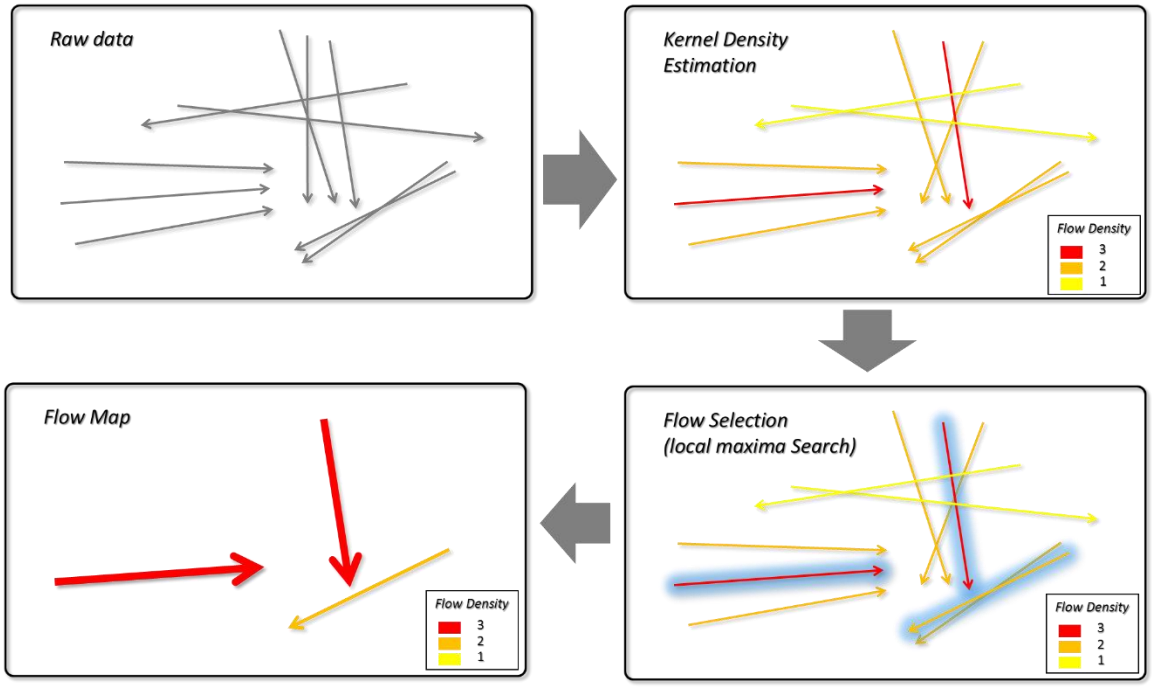

Figure 3.1 Illustration of the overall methodology.

3.4.1 Flow based Kernel Density Estimation Model

Kernel density estimation is a statistical technique for removing spurious data variation and estimating a reliable density value of the observed data points (Silverman 1986). The kernel density estimator defined as following:

$$f_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_H(x_i - x)$$

Where $x_1, x_2, \ldots, x_n \in R^d$ is a set of sample data and $x_i - x$ represents the distance from data point to test point. K is the kernel function and H is a $d \times d$ bandwidth matrix. The kernel function can be Epanechnikov, Triangular, or Gaussian. The choice of the kernel function does not change the smoothing result significantly, while the bandwidth parameter H is key parameter that determine the flow density estimation.

In this study, we applied the kernel density estimation method on the OD trip dataset to investigate mobility patterns. Let $T = (T_1, T_2, \ldots, T_n)$ be an OD trip dataset has n observations. $T_i = (o_x, o_y, d_x, d_y)$ is an OD trip, which start from point O (o_x,o_y) to point D(d_x,d_y). In this research, we treat each OD trip as one data point in 4D space.

The bandwidth matrix induces an orientation is a basic difference between multivariate kernel density estimation from its univariate analogue. In this study, we set the bandwidth matrix as 4 dimensional positive scalars times the identity matrix, which assume the kernels have the same weight in all of the four dimensions.

$$H = \begin{pmatrix} h & 0 & 0 & 0 \\ 0 & h & 0 & 0 \\ 0 & 0 & h & 0 \\ 0 & 0 & 0 & h \end{pmatrix}$$

The bandwidth for origin and destination can be different and adaptive to location-specific characteristics such as population density or location type, which can be implemented by setting the values of the bandwidth matrix. In this paper, we use the same neighborhood size for origin and destination for a given scale.

The selection of bandwidth has a strong influence on the result of density estimation. Given a dataset, too small bandwidth may cause under smoothed estimation while too large bandwidth may cause over smoothed estimation. The most common optimality criterion used to select bandwidth such as MISE (mean integrated squared error) or AMISE (Asymptotic MISE) are unable to be used directly since the true density distribution of the dataset is unknown. The Silverman's rule is an approximation method based on the assumption that the underlying density being estimated is Gaussian, which suggest $\sqrt{H_{ii}} = (\frac{4}{d+2})^{\frac{1}{d+4}} n^{\frac{-1}{d+4}} \sigma_i$ where σ_i the standard deviation of i-th variable of the samples and n is the number of the observations. Hence, data-based bandwidth selection methods(Jones,

Marron and Sheather 1996, Sheather and Jones 1991) are practical and useful for a wide range of data set.

In our scenario of flow map generalization, the extent of map is an important factor for us to choose appropriate bandwidth. It is practical to choose a bandwidth proportional to size of map. For example, set h similar to 1/10 of the map length can naturally discover flow patterns among regions which area similar to 1/100 of the map areas. Besides, different bandwidths represent varies actual meaning in the geographic space. For instance, a neighborhood size of 20 meters may be suitable for recognizing a bus stop, while a much larger neighborhood size, e.g. 500 meters, maybe needed for large locations such as a stadium. Different parameter settings can detect flow patterns at different scales. The bandwidth selection is highly depends on the requirement of the applications.

To sum up, the selection of bandwidth suggested by the data-based bandwidth selection method or expert who understand the context of the application are both applicable for our method. In this paper, we combine our method with the interactive map by fix the ratio between smooth bandwidth and map extent, and then explore flows patterns by change the zoom in level of interactive map. Using a small scale to discover general flow patterns at a higher level while a larger scale can reveal local flow patterns among smaller areas. A user may start from the global level with a large neighborhood size to obtain an overview of global patterns and then zoom in on a local region (with the neighborhood size scaled down accordingly) to explore patterns with more local details.

3.4.2 Flow Selection and Generalization

In traditional map generalization, the cartographer is responsible for selecting the most necessary elements and suppressing the unimportant details to reduce data complexity

and achieve map clarity and balance. It is a challenging problem to automate such a selection process, which is of critical need for dealing with big dataset and supporting interactive data exploration across scales. In this paper, we designed a new flow map generalization method with local maximal search, which based on the kernel density estimation result conducted in section 3.4.1

The essence of the flow map generalization idea is to generalize the generated density surface into several 4D local maxima points, and those 4D points can be visualized as flowlines on 2D flow map. The generalization process need to (1) preserve overall patterns by selecting important and representative flows and (2) maintain the clarity of flow map by only selecting a small set of flows that are not too close to each other. For the first objective, we consider a flow with higher smoothed density more important. To achieve the second objective (i.e., avoiding the cluttering problem), we ensure that selected flows are not within each other's neighborhood, defined by a search radius. Imagine that the first step of smoothing generates a smoothed density surface in 4D spaces, where the peaks represent a local cluster (or hot spot) of flows close to each other. The selection method will ensure that all selected flows represent local peaks. We adopted a greedy search strategy to achieve to achieve these objectives. The selection algorithm is presented below (Algorithm 3.1).

Algorithms 3.1. Flow Selection

Input: Smoothed flows T= $\{T_f'\}$;
Neighborhood search radius r;

Output: A set of selected flows S={Ts};

Steps:

```
S = Ø;
FOR EACH Flow T'_f ∈ T:
        isLocalMax = false;
        currentFlow = T'_f;
        WHILE  (isLocalMax == false)
                neighborFlows  = searchFlowNeighbor(currentFlow, r);
                maxFlow = findMaxFlow(neighborFlows);
                IF  (maxFlow == currentFlow)
                        IF  currentFlow.selected = false
                                S = S ∪ {currentFlow};
                                currentFlow.selected = true;
                        END IF;
                        isLocalMax = true;
                END IF;
                ELSE
                        currentFlow = maxFlow;
                END ELSE;
        END WHILE;
END FOR;
```

The method *searchFlowNeighbor (flow, r)* returns a list of neighboring flows for a given flow. Neighborhood search can be supported efficiently by a spatial index such as R*- tree. The method *findMaxFlow(neighborFlows)* returns the flow with the highest smoothed value within a list of flows. For each original flow, the selection algorithm starts from the flow, move to the maximal flow within its neighborhood, then find the maximal flow within its new neighborhood, until the current flow is the local maximum within its flow neighborhood. This way, a representative flow is identified for each original flow. The only parameter in this process, i.e., the search radius *r*, controls the pattern resolution while the smoothing bandwidth (in the smoothing step) controls the pattern scale. A larger search radius select less flows (by jumping over small peaks to reach higher peaks) and gain a more simplified flow map with less details. However, the search radius should vary within a certain range according to the scale. Otherwise, the map may become too detailed or too empty.

3.4.3 Interactive Multi-Scale and Multi-Resolution Flow Mapping

In this paper, we proposed an exploration for an interactive multi-scale flow map. The idea is setting different smooth bandwidth and search radius according to scale. Different smooth bandwidth and search radius can gain patterns at different resolution, then we employ human interaction to switch the flow map between scales. For example, using left flow map in Figure 3.4 as global flow map and using the right map in Figure 3.6 as a zoom in flow map. A user may need to understand the general patterns at a small scale, and then zoom in to a local area to get more details.

Combining the flow kernel density estimation and the flow selection steps, there are two key parameters: the smoothing bandwidth and the search radius, which together controls the pattern scale (from local flow patterns to global flow patterns) and pattern resolutions (how much details to map for a certain pattern scale). Intuitively, a small smooth bandwidth and a small search radius can capture local flow patterns with more details (i.e., high spatial resolution). However, the flow map may have the cluttering problem when too many flows selected. Therefore, given a certain scale, it is a tradeoff between map clarity and map resolution. Normally, the smoothing bandwidth and search radius highly depends on the dataset. The selection of smoothing bandwidth and search radius is according to the interactive visual experience and data distribution. Similar to the selection of bandwidth in kernel density estimation, too large or too small bandwidth may cause under smoothed or over smoothed problem. The search radius may affects the represented flow selection, the local maximal flow but relatively low value compare to

nearby local maximal flow will be ignored with larger search radius. It is practical to implement real-time interactive software to let user determine these two parameters.

3.5 CASE STUDY

3.5.1 Data: NYC Taxi Trips

The new York taxi dataset we used in this study covers all taxi trips of the yellow cabs operating in New York City from Year 2009 through 2015, which contains more than one billion (1,179,731,355) taxi trips. In this case study, we applied our proposed method on New York taxi dataset to investigate the human mobility patterns.

Taxi trips are valuable sensors of city life (Ferreira et al. 2013), which provide insights into many aspects of urban, from transportation planning to human mobility. In this case, every taxi trip has a time stamped origin GPS point and a destination GPS point, which also carries the information about the moving trends. The main challenge of analyzing this dataset is the volume of the data. We partition the study area into 1321*1940 grids, with each grid size is 5 meters. Two maps in Figure 3.2 show the complexity and distribution of the original dataset. The left map in Figure 3.2 shows net flow volume (in – out) for each grid, and the right map in Figure 3.2 shows sum flow volume (in + out) for each grid. Figure 3.3 shows origins and destinations of a subset of taxi trips.

It is impossible to visualize the flows in the dataset individually. The flow map becomes extremely cluttered even for this relative small (hundreds) sample data set, and it is difficult to discover any insights on the movement patterns of people. The left map in Figure 3.2 shows the net flow volume of all the origins and destination points. The locations in blue color recognized as transportation hubs such as: Penn Station, Grand Central and

other subway stations, since taxi is a supplement transportation for other public transit. The downtown area and convention center with red color represent the main destination for people in the busy morning.

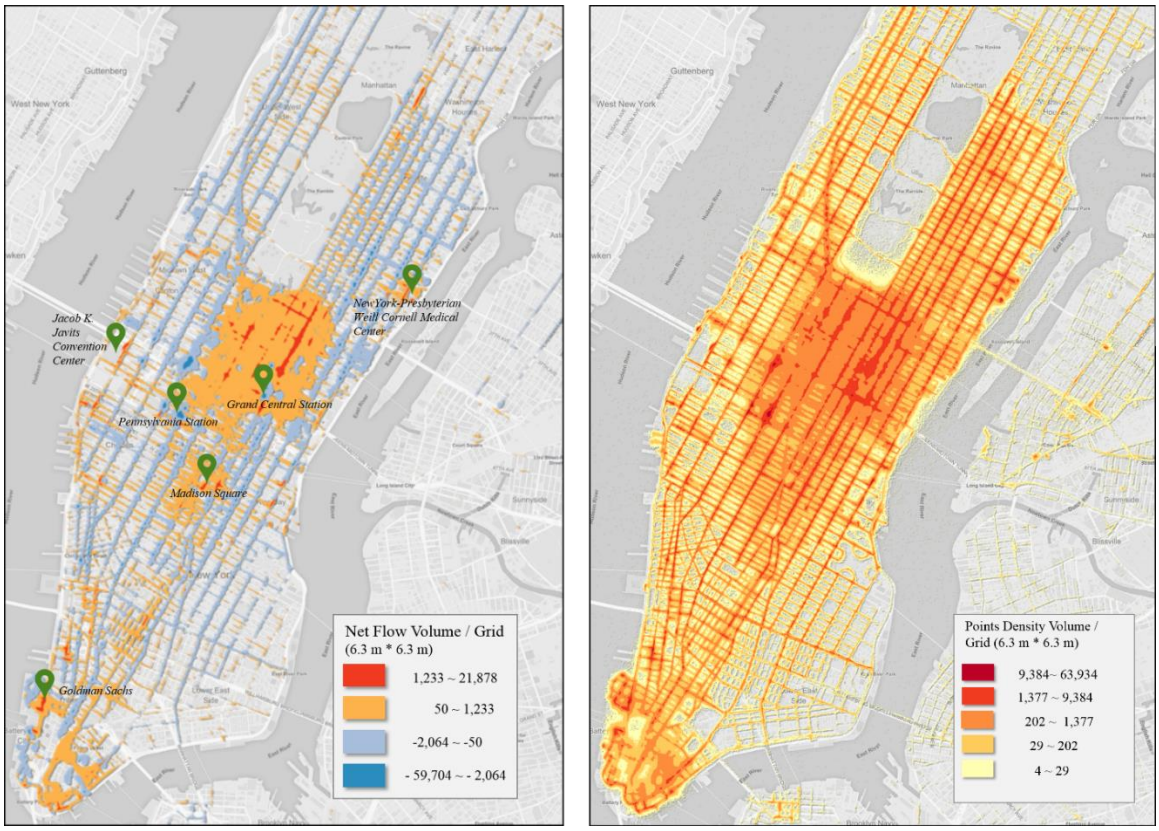

Figure 3.2 Distribution of the origin and destination of taxi trips. The left map shows net flow volume (in – out) for each grid, and the right map shows sum flow volume (in+out) for each grid.

It is impossible to visualize the flows in the dataset individually. The flow map becomes extremely cluttered even for this relative small (hundreds) sample data set, and it is difficult to discover any insights on the movement patterns of people. The left map in Figure 3.2 shows the net flow volume of all the origins and destination points. The locations in blue color recognized as transportation hubs such as: Penn Station, Grand Central and

other subway stations, since taxi is a supplement transportation for other public transit. The downtown area and convention center with red color represent the main destination for people in the busy morning.

Despite the midtown area, it is hard to tell which area has more arrival than departures. (More departures locates at the avenue and more arrivals locates at the streets.) Location measure is an efficient complimentary way to understand O-D flow data, but it also has the problem of scale. In Figure 3.4, the upper map shows the smoothed locational measure and highlights hotspot for departures (in blue) and arrivals (in red). Compare to the right map, the left map use 100 meters as the smooth radius for kernel smoothing, and the right map use 300 meters as the smooth radius. These two maps demonstrate different locational measures at various spatial scales. These two maps also work as the background to verify the flow maps produced by our method in the following sections.

3.5.2 Flow Map Generalization

Figure 3.5 is the generalized flow map within this study area. 80 flows in the map represent the most typical flow patterns in the workday morning. As we use the uniform model, every O-D flow count as weight 1 in the flow density estimation model. To simplify, the strongest flow in the map means there are 532,264 taxi trips between two circles with 300 meters radius during morning hour in past seven years.

3.5.3 Selection Parameter Configuration

From the flows with smoothed value, we use the flow selection algorithm presented in Section 3.4 to select local maximal flows. To examine the sensitivity of the selection results to the search radius configuration, the selection results with search radius of 200 meters, 300 meters and 300 meters are shown in Figure 3.7 and Figure 3.8. Comparing

these three maps, we can see that: (1) larger search radius selects less flows and improve the clarity of flow map, because larger search radius can jump out from the local maximum which the value is relatively low.

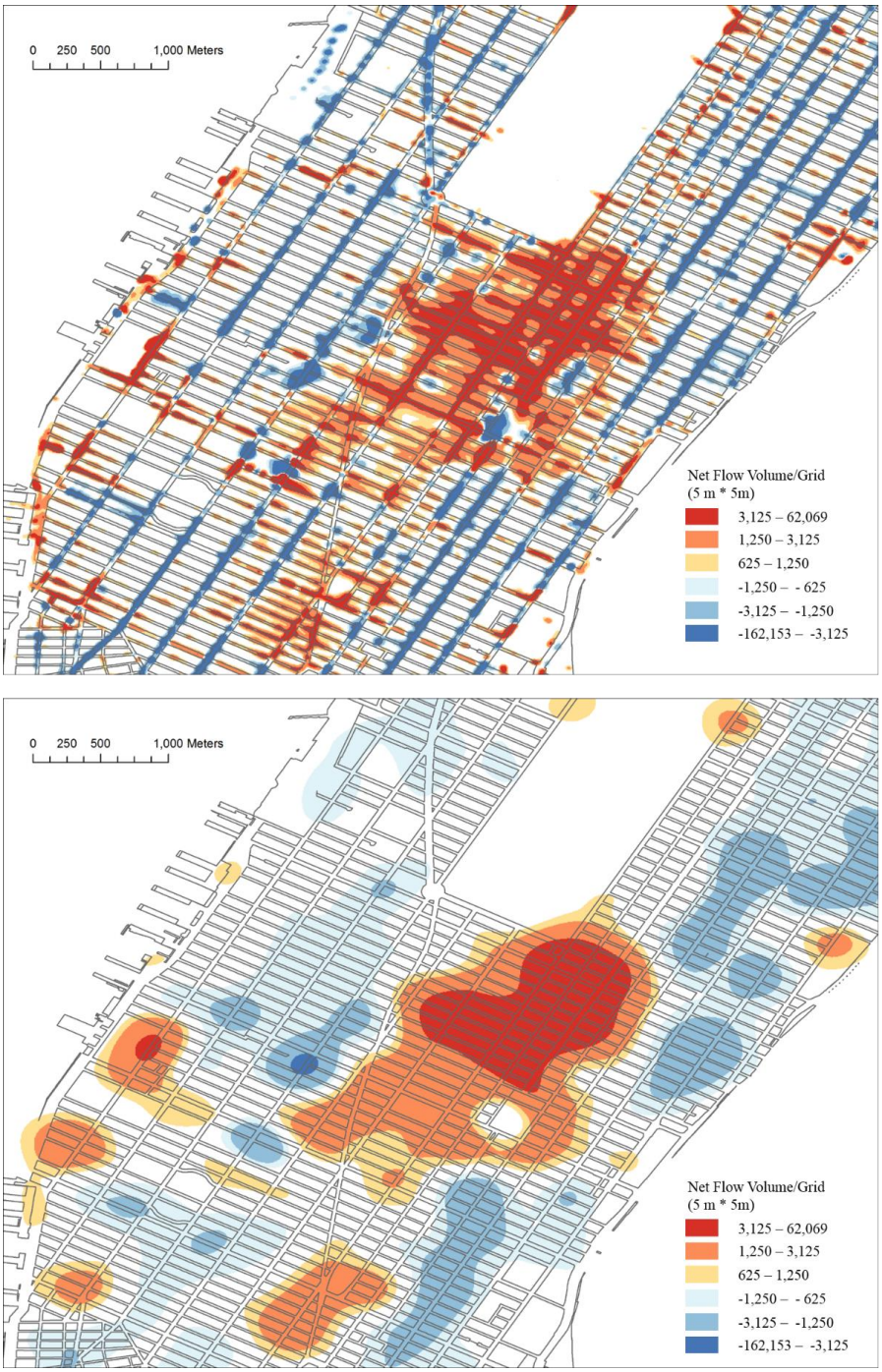

Figure 3.3 Illustration of scale problem. Location measure maps at different spatial scales.

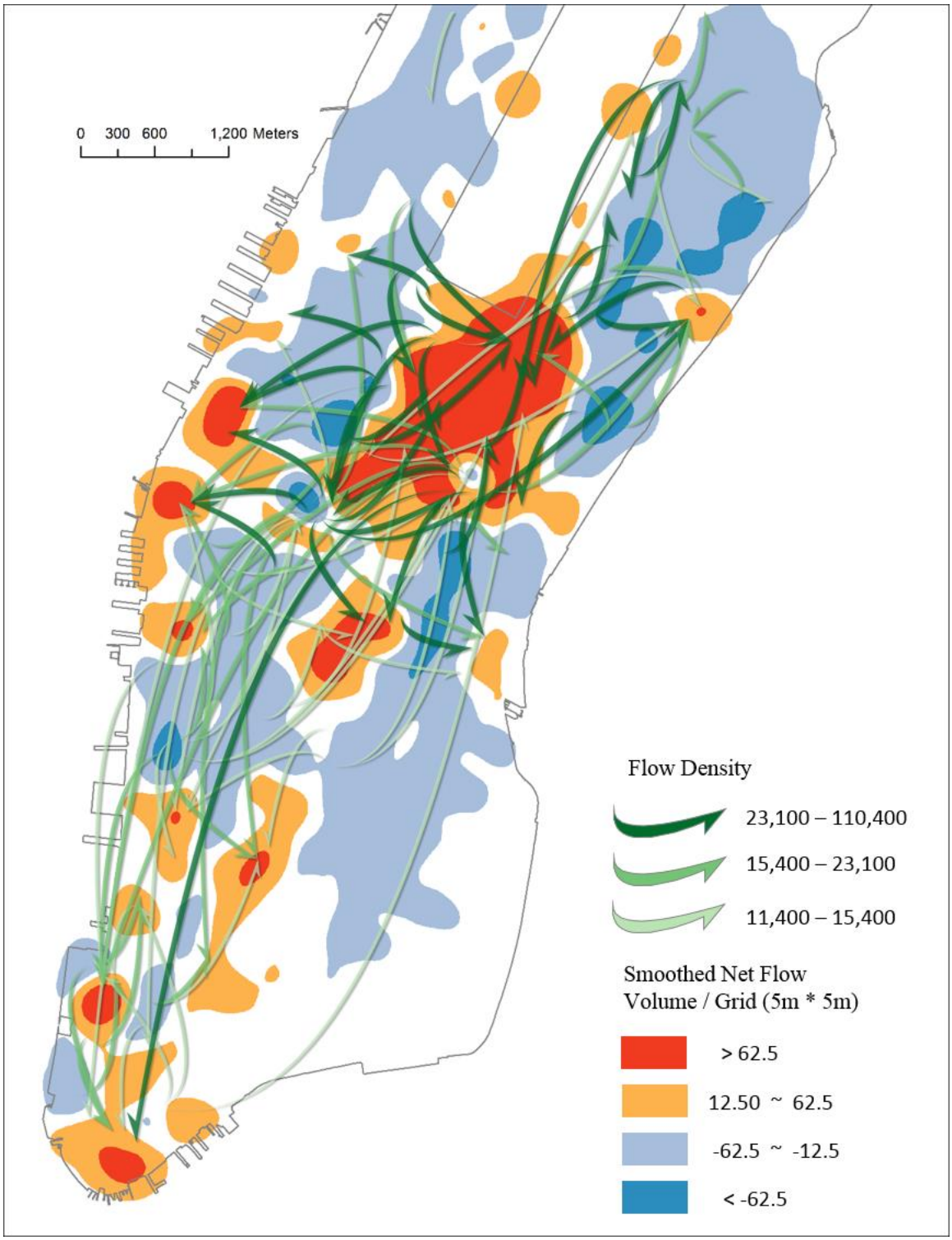

Figure 3.4 Generalized Flow Map for 110,076,167 taxi trips, which occurs on workday morning (7am – 9am) from 2009 to 2015.

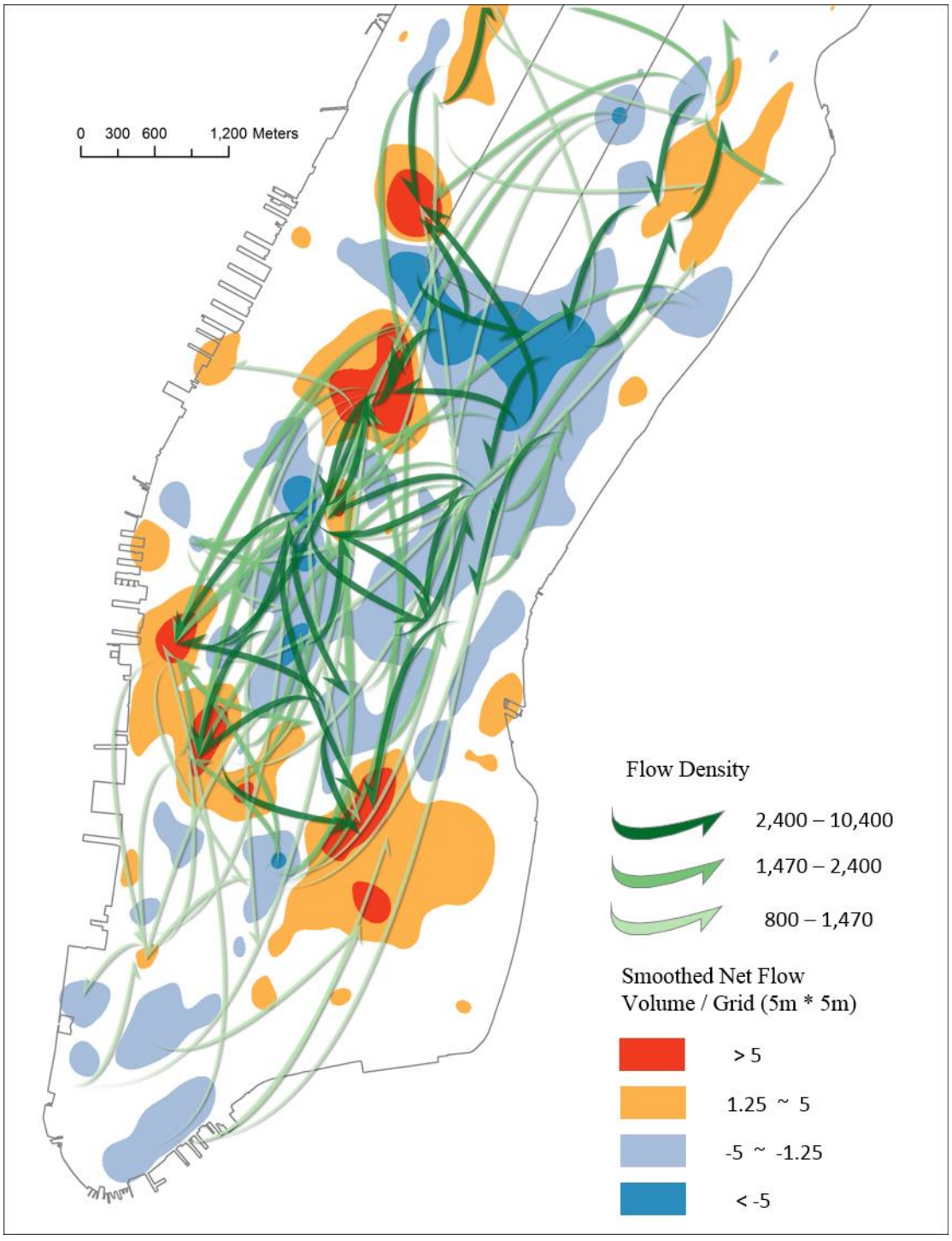

Figure 3.5 Generalized Flow Map for taxi trips, which occurs on Saturday (6pm – 7pm) from 2009 to 2015.

However, it will lose more details of the flow patterns. (2) The selected flow set by larger search radius is the subset of the selected flow set with smaller search radius. In our

interactive flow map design, we provide users an interface like a scroll bar to input the search radius, and the flow map can real time response to user's input. There is no optimal solution for the search radius selection, and it totally depends on the user's preference, information abundance or map clarity.

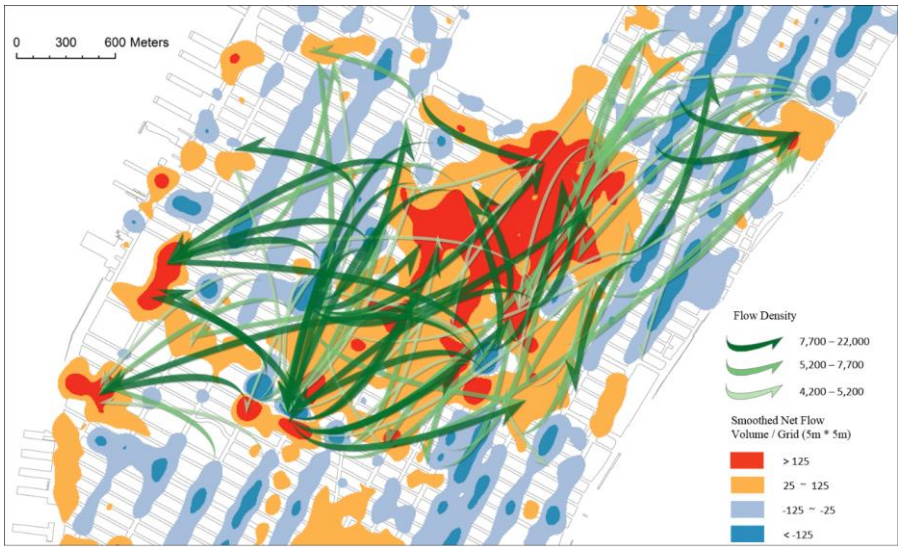

Figure 3.6 Flow map for downtown area. Smooth radius is 100 meters and selection radius is 200 meters.

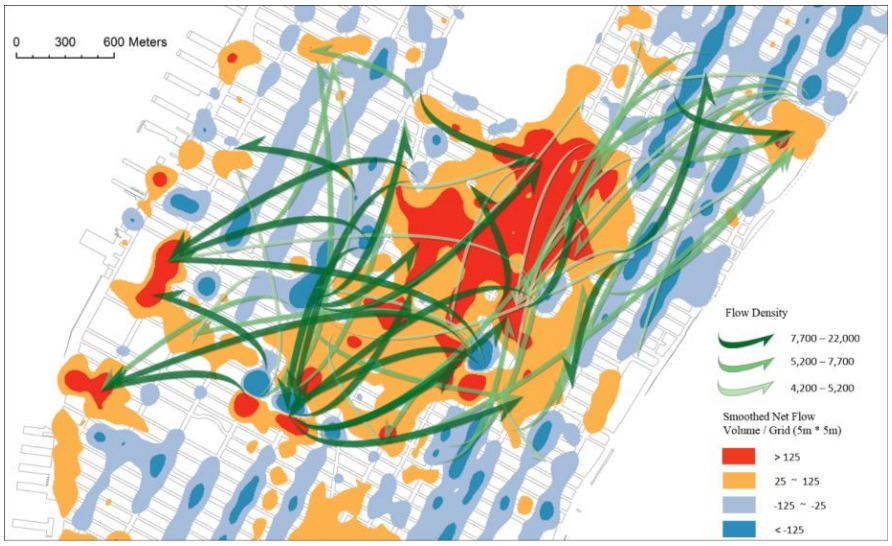

Figure 3.7 Flow map for downtown area. Smooth radius is 100 meters and selection radius is 200 meters.

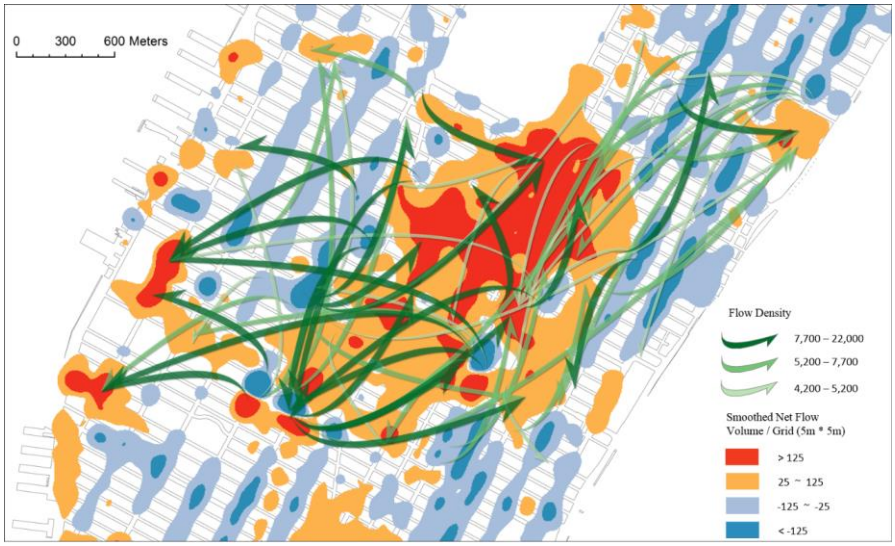

Figure 3.8 Flow map with 100-meter smooth radius and 300-meter selection radius.

3.5.4 Multi-scale Interactive Flow Mapping

Figure 3.9 has two flow maps for Manhattan area as we showed before. In the left map, we use 300 meters as the smooth bandwidth and 300 meters as the search radius for select reprehensive flows. There are 80 flows which flow value greater than 8000 shown in the left map. In right map, we use 100 meters as the smooth bandwidth and 200 meters as the search radius for select flows. The right flow map is a zoom in perspective for the region selected in the left map. Comparing these two maps, the patterns are similar, but the right flow map carries more detailed by zoom in to this region.

3.6 CONCLUSION

This paper presents a multi-scale flow mapping method which is scalable for mapping large origin-destination flow data at different scales. It can be used for interactive analysis of flow patterns. The key components are a flow map generalization method based

on kernel density estimation. The smoothing method requires the input parameter to define kernel bandwidth, which supports the implementation of multi-scale flow map. The configuration of kernel bandwidth is critical and influences the smoothing results dramatically. Different bandwidths reveal the patterns in different scales. Larger bandwidth can discover general patterns from large area to area, while smaller bandwidth can discover specific patterns from small area to area.

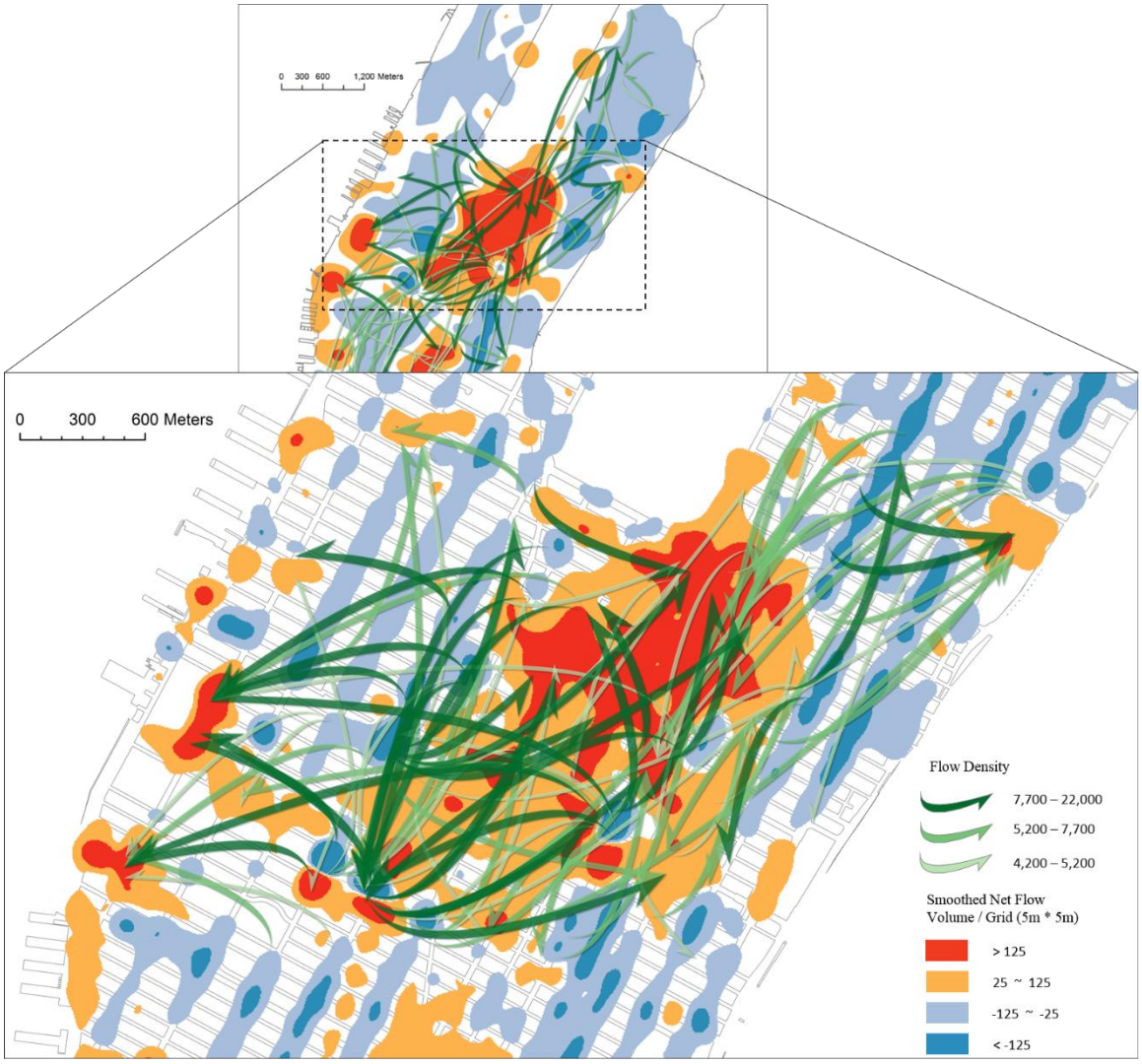

Figure 3.9 Flow map of taxi trips in the Manhattan area at different spatial scales. The right flow map is a zoom in perspective for the region selected in left flow map, only the flows which origin or destination located within this area showed.

The selection method is to find a subset of flows that can represent the major flow patterns in the data. The parameter of search radius determines the extent of map generalization: larger search radius selects less reprehensive flow and gets clearer flow map. Meanwhile, it will lose more details and resolutions of flow patterns. We carried out a case study which is the taxi trips in New York City to evaluate the usefulness of our proposed approach. Results show that the proposed method can effectively discover patterns in different scales and resolutions.

Future work is needed to better balance and automatically determine the appropriate level of flow smoothing bandwidth and selection search radius to optimize the flow map layout. This will require both intelligent algorithms and interactive human inputs. The effectiveness of the proposed approach needs further implementation with interactive software. Of course, supporting innovative user interactions will add additional benefits and will be explored in future work.

CHAPTER 4 URBAN EVENT DETECTION WITH BIG DATA OF TAXI OD TRIPS – A TIME SERIES DECOMPOSITION APPROACH

4.1 ABSTRACT

Big urban mobility data, such as taxi trips, cell phone records, and geo-social media check-ins, offer great opportunities for analyzing the dynamics, events, and spatiotemporal trends of the urban social landscape. In this article, we present a new approach to the detection of urban events based on location-specific time series decomposition and outlier detection. The approach first extracts long-term temporal trends and seasonal periodicity patterns. Events are defined as anomalies that deviate significantly from the prediction with the discovered temporal patterns, i.e., trend and periodicity. Specifically, we adopt the STL approach, i.e., seasonal and trend decomposition using LOESS (locally weighted scatterplot smoothing), to decompose the time series for each location into three components: long-term trend, seasonal periodicity, and the remainder. Events are extracted from the remainder component for each location with an outlier detection method. We analyze over a billion taxi trips for over seven years in Manhattan (New York City) to detect and map urban events at different temporal resolutions. Results show that the approach is effective and robust in detecting events and revealing urban dynamics with both holistic understandings and location-specific interpretations.

4.2 INTRODUCTION

Big urban mobility data, such as taxi trips, cell phone records, and geo-social media check-ins, offer great opportunities for the analysis and understanding of the dynamics, events, and spatiotemporal trends of the urban social landscape. Such data can potentially enable both real-time monitoring and long-term analysis of geo-social dynamics. However, there is a lack of effective methodologies to characterize urban patterns, quantify trends, and simultaneously detect events or anomalies from big and long-term urban mobility data. Previous research on urban mobility data analysis includes the detection of social events such as human gatherings with a density analysis of mobile phone usage (Calabrese et al. 2010, Candia et al. 2008), the examination of responses to earthquakes from twitter data (Sakaki et al. 2010), and the analysis of land use transition and anomalies with taxi trajectories (Pan et al. 2013). Existing methods for urban event detection do not usually separate trends from periodicities, so the event detection process can be influenced by hidden trend changes. Temporal trend and periodicity often change over space and time and urban dynamics may exhibit patterns also occurring at different spatial scales and temporal granularities.

It is a challenging problem to effectively extract long-term trend, periodicity, and events from high-resolution and big urban mobility data, which may have billions of GPS points and trips over years. Compared with the commonly used mobile phone data and social media check-ins, taxi trip data have higher resolutions in both geographic space (about 5 m accuracy) and time, which has the potential to detect events with high accuracy. Cell phone data has a much lower spatial resolution, depending on the density of cell phone towers, normally ranging from hundreds of meters to kilometers. Social media data, on the

other hand, has a low and uneven temporal resolution, where some users may send multiple tweets within a few minutes while some other users only send one tweet in several days.

In this research, we present a new approach for detecting and mapping urban events with big data of taxi OD trips based on time series decomposition and outlier detection. The approach can distinguish different temporal patterns (e.g., long-term trends and seasonal periodicity), define and detect events as anomalies deviating from the patterns, and map the dynamics of events over space and time with high accuracy and resolution. Specifically, our approach uses the STL method (Cleveland et al. 1990) i.e., Seasonal and Trend decomposition using LOESS (locally weighted scatterplot smoothing), to decompose the time series for each location into three components: long-term trend, seasonal periodicity, and the remainder. Urban events are extracted with an outlier detection method from the remainder component, which has removed both seasonal periodicity and long-term trend. We analyze over a billion taxi trips for over seven years in Manhattan (New York City) to detect and map urban events at different temporal resolutions. Results show that the approach is effective and robust in detecting events and revealing urban dynamics with both a holistic understanding and location-specific interpretations.

4.3 RELATED WORK

Event or anomaly detection from spatial-temporal data has been extensively studied, such as detecting extreme precipitation events (Wu et al. 2010), outliers in meteorological data (Lu and Liang 2004) and agriculture data (Lu, Carbone and Gao 2017), disease outbreaks (Kulldorff 1997) terrorism outbreaks (Guo et al. 2012), and spatiotemporal

events in social media. Most of existing methods for spatiotemporal event detection are based on density-based clustering or spatiotemporal scan statistics, which do not take into account long-term temporal trends and periodicity. Another group of methods for event or anomaly detection in time series analysis is based on a regression model, which estimates its parameters by data fitting and extracting anomalies from the residuals, i.e., the difference between the observed value and the value forecast by the regression model (Bianco et al. 2001). Model-based approaches in general are confirmatory analyses and therefore cannot discover unknown forms of patterns and may suffer from the over-fitting problem.

Unlike model-based approaches, there are numerous methods for exploratory time analysis. The simplest one is to use a moving average window to smooth the time series and detect long-term trends, which does not explicitly recognize periodicity patterns and anomalies. To detect periodicity, the Discrete Fourier Transform (DFT) or Wavelet Transform methods can be used, which however cannot reveal long-term trends or anomalies. To recognize and separate different types of patterns in a time series, decomposition approaches are often used. The STL method (Cleveland et al. 1990), i.e., seasonal and trend decomposition using LOESS (locally weighted scatterplot smoothing), is a data driven exploratory method that can decompose a time series into trend, seasonality (periodicity), and remainder components. Different applications of STL may focus on different components, e.g., climate studies may focus on exploring the trend and economists are often interested in the business cycle (seasonal component). The focus of our research is on the remainder component, which can be used to detect events. For example, Chae et al. (Chae et al. 2012) integrate thematic modeling and STL to visualize

social media anomalies and Hafen et al. (Hafen et al. 2009) use STL to monitor health data and detect disease outbreaks. Since STL is essentially a smoothing-based approach, it is not suitable for detecting the change point (Verbesselt et al. 2010b, Verbesselt et al. 2010a).

Big mobility data, such as geotagged social media, mobile phone usage and taxi trips, have become increasingly available and offer unprecedented opportunities to understand the geographic and social dynamics (Liu et al. 2015). Social media check-ins, often with real-time data feeds, have been widely used in event detection (Chae et al. 2012, Dong et al. 2015a, Sakaki et al. 2013, Sakaki et al. 2010). Recently, there has been an increasing interest in exploring both the temporal and spatial dimensions in social media data to extract and understand events at different spatial-temporal scales (Chen and Roy 2009, Dong et al. 2015a, Rattenbury et al. 2007). Mobile phone data have also been widely used in urban activity monitoring (Calabrese et al. 2010, Calabrese et al. 2014, Ratti et al. 2006), event detection (Candia et al. 2008, Dong et al. 2015b, Traag et al. 2011), population density mapping (Deville et al. 2014), and tourism management (Ahas et al. 2007, Ahas et al. 2010). A more complete review on mobile phone data analysis can be found in Calabrese, Blondel, and Ferrari (Calabrese et al. 2014).

Mobile phone and social media data have the advantage of high penetration and demographic coverage. However, social media and mobile phone data have uneven and often low spatiotemporal resolutions. For example, the Call Detail Record (CDR) data, a type of mobile phone data, records the user location when a call or text message is made or received, and the location recorded is the location of the cell tower. Therefore, the spatial resolution of mobile phone data are not very high and vary with the distribution of cell towers, ranging from hundreds of meters to kilometers. To improve the usefulness of

mobile phone data, a number of approaches have been developed that either use probabilistic location inference models to enhance location accuracy (Traag et al. 2011) or combine multiple data sources to assist application-specific analyses (Calabrese et al. 2011, Liu et al. 2015, Sagl et al. 2012).

Taxi trip data have high spatiotemporal resolution and are suitable for extracting urban events with high accuracy (Calabrese et al. 2010, Zhang et al. 2015). Taxi trip data can be grouped into two kinds: trajectory data (with actual driving routes) and OD trips (with only the origin and destination of each ride). Scholz and Lu (Scholz and Lu 2014) present a method to analyze activity hot spots of urban activities with massive trajectory data. Their method defines an activity hot spot as a location with an extremely large number of activity instances during a certain hour and assumes that the theoretical distribution of activity instances across the study area and through the study time is completely random. As such, the method does not take into account either temporal trends or periodicities in defining events or hot spots. Zhang et al. (Zhang et al. 2015) introduce an event detection method that can consider temporal periodicity (i.e., fluctuation patterns repeated in time), which uses the Discrete Fourier Transformation (DFT) to find the length of periodicity and then define events as deviations from the periodicity. This approach does not consider the long-term temporal trend of activities, particularly for very long time series. For example, taxi pickups or drop-offs at a specific location may gradually (or quickly) increase or decrease if the land-use type of the location changes, which should also be considered in event detection other than periodicity.

Different from existing approaches, our approach considers both long-term trends and seasonal periodicities in defining events with big spatial mobility data. We use the STL

method (Cleveland et al. 1990) to decompose time series into three components (long-term trend, seasonal periodicity, and the remainder) and extract events from the remainder component.

4.4 DATA

The data used in this research cover all taxi trips of the yellow cabs operating in New York City for seven years, from 2009 through 2015. The data has a total of over one billion (1,179,731,355) taxi origin-destination (OD) trips. Yellow cabs represent the majority of taxi cars in the Manhattan area for the seven-year period. The green cabs started operation in August 2013 and are only permitted to pick up passengers to the north of the 110th Street (northern Manhattan) or in the outer-boroughs of New York City. Other riding-share options such as Uber and Lyft became available after 2015. This big data set of taxi trips can provide comprehensive information for understanding urban dynamics. Each taxi OD record has a number of fields, including pick-up and drop-off dates and times, pick-up and drop-off locations, trip distance, itemized fares, rate type, payment type, and passenger count. In this study, we primarily focus on the location and date/time of pickups and drop-offs.

Figure 4.1 shows a density map of taxi pick-ups in the Manhattan area, which is divided into a set of grid cells (each cell is 20 by 20 m in size) and the total count of taxi pick-ups (for seven years) for each cell is mapped. To showcase the time series at different scales, one specific grid cell (200 West Street, the address for the Goldman Sachs Tower) is selected and its time series at three temporal resolutions (i.e., weekly, daily and hourly) are shown in Figure 4.2. As seen in the plots, temporal patterns are complex and can

involve both long-term trends and periodicities (repeated patterns) at various scales, as well as noise and outliers. The Goldman Sachs Tower was completed and put to use at the end of 2009, which completely changed the overall time series. We can also notice that there was an anomaly in mid-September of 2010, visible in the daily and hourly time series plots. Existing event detection methods may consider periodicities but often ignore the overall trend (e.g., declining, increasing, or more complex changes), which is a severe limitation particularly for event detection with long-term time series that span across many years.

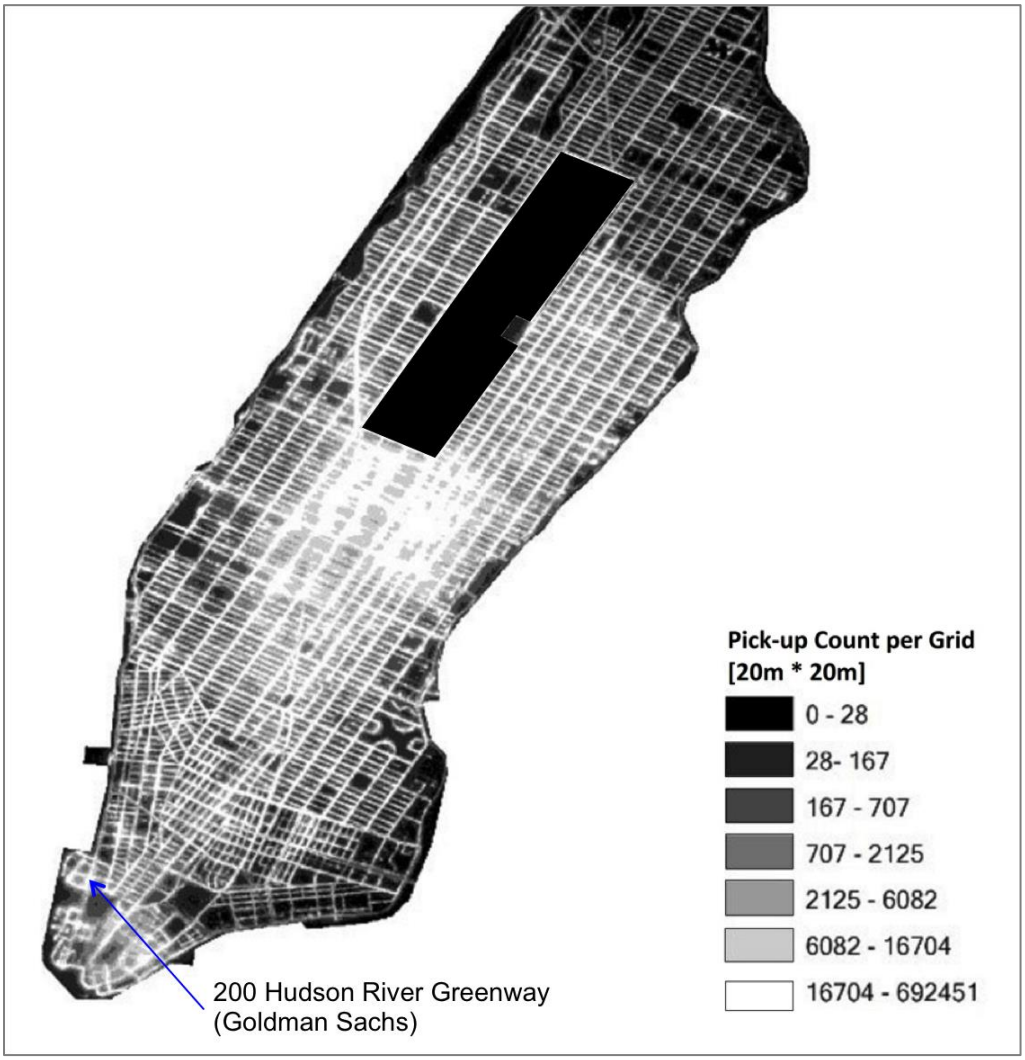

Figure 4.1 Taxi pick-up counts per grid cell (20m*20m) for seven years in the Manhattan area.

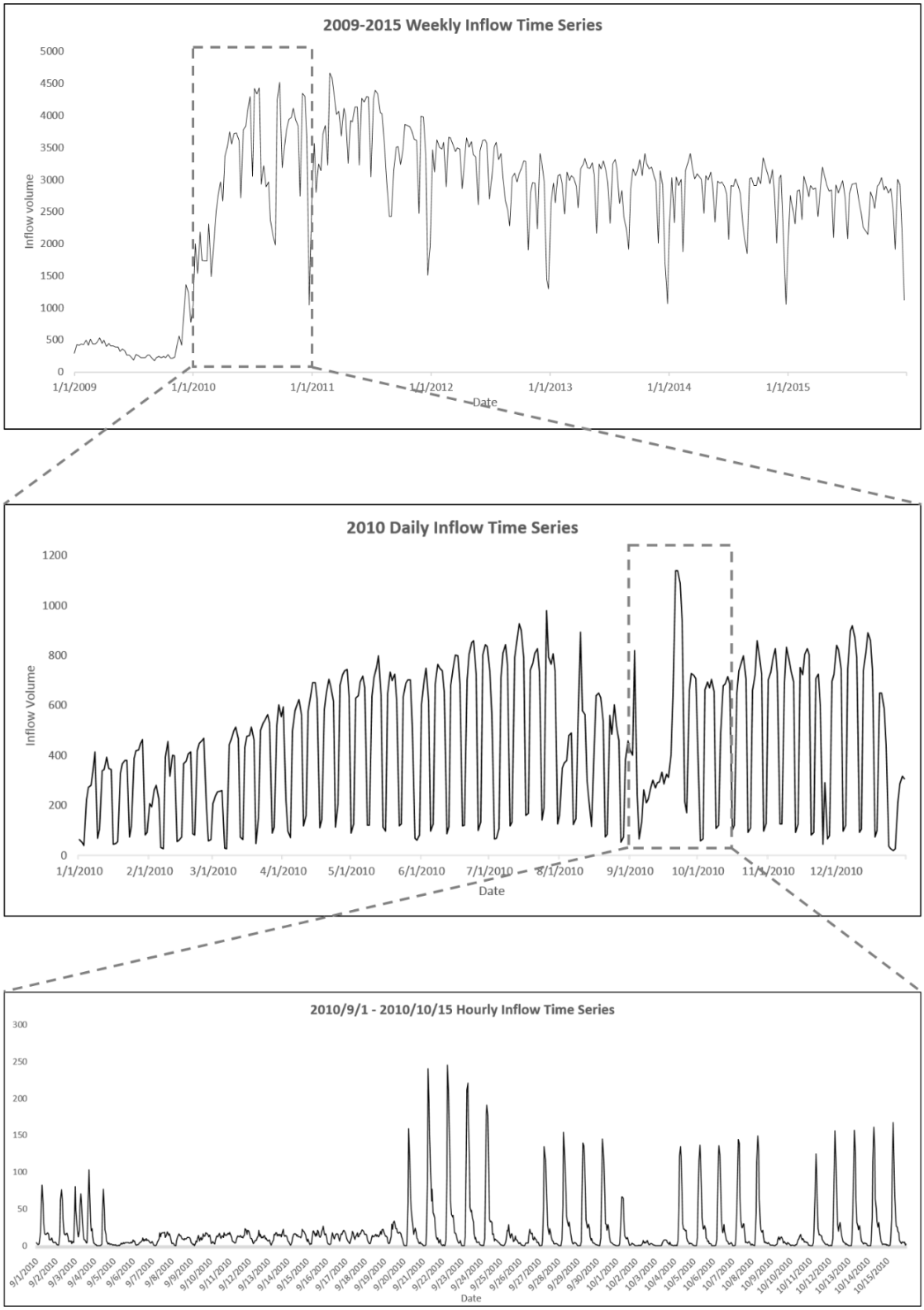

Figure 4.2 The time series for the grid cell at 200 West Street (Goldman Sachs Tower), at three different temporal resolutions, i.e., weekly, daily, and hourly.

4.5 METHODOLOGY

In this article, we propose a new approach for urban event detection with billions of taxi trips based on the STL method (Seasonal and Trend decomposition using Loess) (Cleveland et al. 1990) and an outlier detection method. The approach consists of three steps.

First, partition the study area into a set of grid cells (e.g., 20 by 20 m grid cells as in the case study) and construct a time series for each grid based on the taxi pickups or drop-offs in the neighborhood (e.g., a 50 m buffer) of each cell over time.

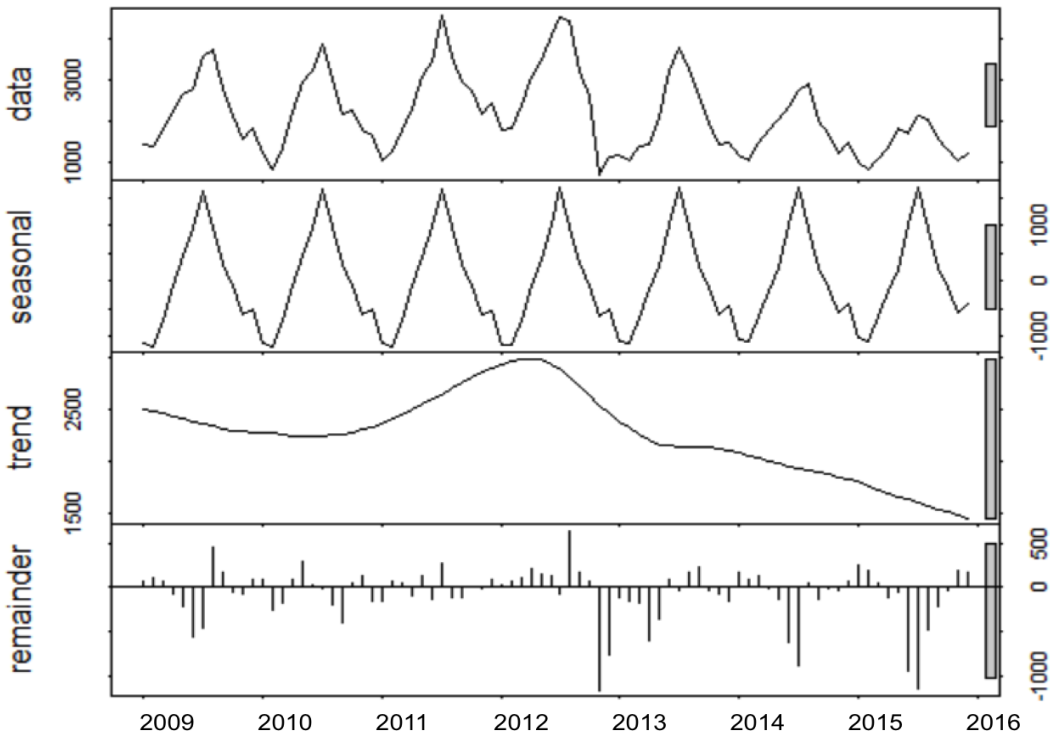

Figure 4.3 An illustrative example of time series decomposition with monthly taxi drop-offs over seven years (2009 – 2015) for a selected location in Manhattan, NYC. From the top to the bottom are: (1) original data; (2) the seasonal periodicity component, (3) the long-term trend component; and (4) the remainder component, from which anomalies (events) will be extracted

Second, decompose each time series into three components: long-term trend, seasonal periodicity, and a remainder component, using the STL method (Figure 4.3). This decomposition step is critical because events in time series are often embedded in and complicated by long-term trends and seasonal periodicities (here "seasonal" is a broad term that represents repeated patterns of any time interval).

Third, extract anomalies (events) with the remainder component, which has removed the long-term trend and seasonal periodicity.

Figure 4.3 shows an example of the result, which is the time series decomposition for monthly taxi drop-offs at a specific location, using the STL (Seasonal and Trend decomposition using Loess) method. Without decomposition, it is very difficult to quantify anomalies and detect events from the original time series. Events (anomalies) become much clearer in the remainder component after removing the trend and periodicity components. For each data point in the remainder time series, our approach calculates a modified Z-score to measure its potential as an event. The following subsections will present the approach in detail.

4.5.1 Time series decomposition

We use the STL method to decompose a time series into three components: trend, seasonality (periodicity), and remainder, each representing one type of the underlying patterns. The trend component describes a long-term change pattern in the data. The seasonality of a time series is a pattern that regularly repeats with a fixed interval. The remainder component is essentially the remaining variation in the data that cannot be explained by the seasonal and trend components. The decomposition process allows the

user to specify the amount of variation in the trend and seasonal components, and also allows specifying the length of periodicity, which is useful for analyzing time series of different temporal resolutions. The decomposition outcome can be expressed with an additive model: $Y_v=T_v+ S_v+e_v$, where Y_v is the original time series (data) for location v, T_v , S_v , and e_v represent the trend, seasonal and remainder components, respectively. $T_v+ S_v$ is referred as the deterministic or predictable component.

The overall structure and steps of the STL method is shown in Figure 4.4, with an inner loop nested inside an outer loop. The inner loop has six steps to iteratively fine-tune the trend and seasonal components. Each run of the outer loop updates the remainder component and calculates a set of robustness weights, which are used in the next set of inner loop iterations to reduce the effects of extreme values or outlier observations. The STL method is mainly based on the LOESS method, which is a non-parametric regression model based on a k-nearest-neighbor smoothing (Cleveland 1979). The LOESS model is used at multiple places in the STL method, including steps 2, 3, and 6. LOESS can be based on different polynomial models such as a linear or a quadratic model. In our approach, we use the linear model and the main parameter for LOESS is the size of the smoothing window. In the iterative process, a robustness weight for each data point is iteratively learned, to reduce the influence of outlier values by assigning them relatively small weights in fine-tuning trend and periodicity. Robustness weights for all data points are set to 1 at the start point and will be updated in the outer loop according to the remainder component. The LOESS process in the STL method is essentially a type of k-nearest neighbor kernel smoothing in a temporal context.

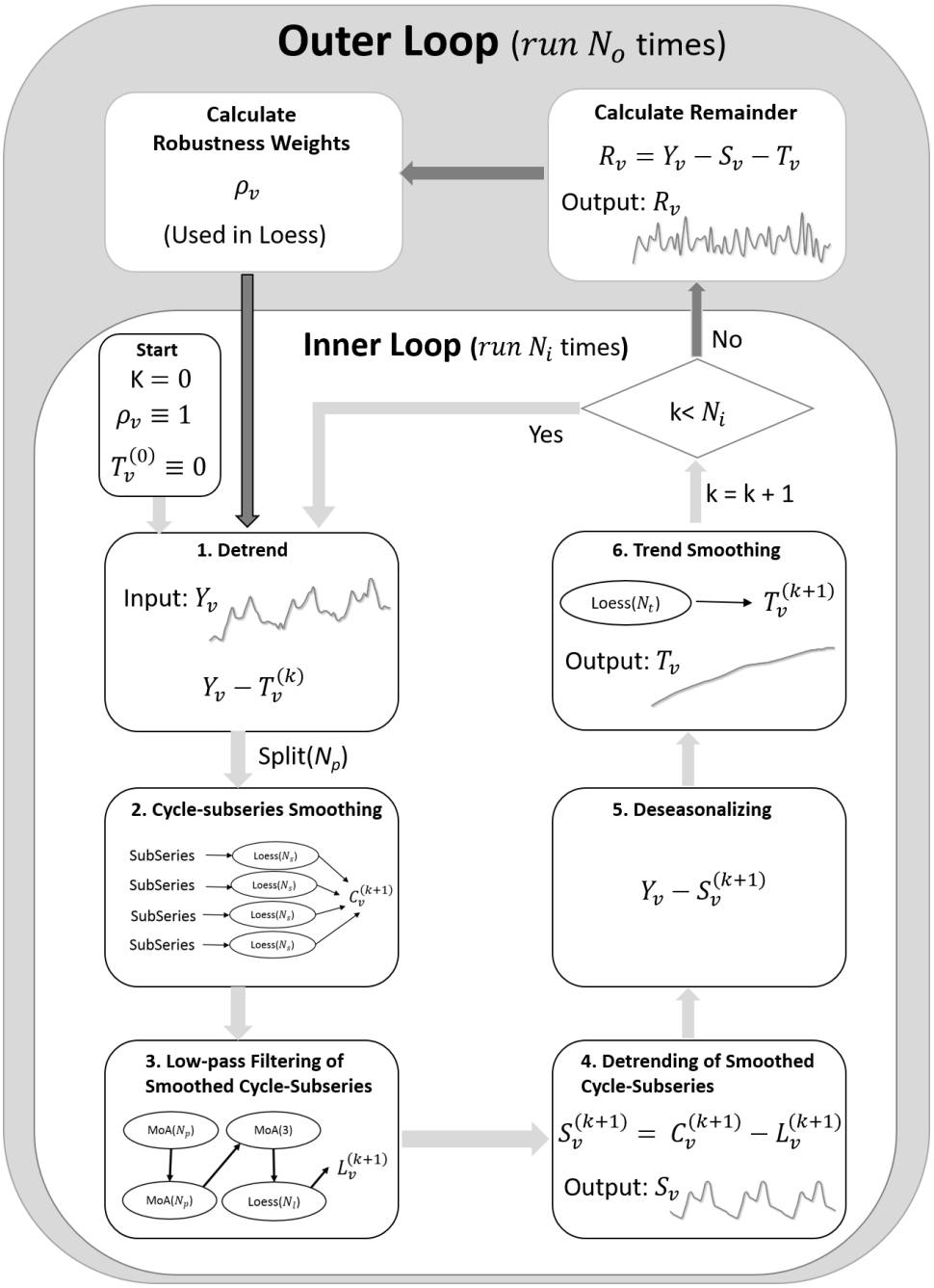

Figure 4.4 Overview of the time series decomposition with STL. Yv is the input time series. Tv, Sv, and Rv represent the outputs: trend, seasonal, and remainder. Split (N_p) is to split the time series into N_p subseries. Loess (N_s) denotes a locally weighted smoothing with a moving window of size Ns. MoA(N_p) denotes a moving average smoothing with a window of size N_p.

The trend component T_v is initially set to a vector of zeros, which will be updated in subsequent iterations. After Step 1 (Detrend) of the inner loop, the detrended time series (i.e., the original time series minus the trend component) is split into Np subseries, where N_p is the number of observations per cycle. For example, if the input data Y_v is a time series of hourly data for seven years with assumed weekly periodicity, then $N_p = 168$ because there are 168 hours per week, and each subseries will be a time series of 365 values because there are 365 weeks in seven years. In other words, each subseries represents the observational data for a specific hour within a week and thus has one value for each week. Similarly, if the input is daily data with weekly periodicity, then $N_p = 7$ and each subseries has 365 values. In Step 2 (cycle-subseries smoothing), each subseries is smoothed with LOESS separately and then the set of subseries are combined to one time-series again (C_v). We will provide detailed discussions on parameter configuration in Section 4.5.3. Figure 4.5 shows the three output components of STL for the daily time series of taxi pickups for seven years (only year 2015 is shown due to limited space) at 200 West Street (i.e., Goldman Sachs Tower).

4.5.2 Event Detection

After the decomposition of the time series for a grid cell, each value R_t in the remainder component is the difference between the observed value Y_t and the expected value (i.e., the trend value Tt plus the seasonal value S_t) for a specific time t. To test the significance of R_t, we again split the remainder component into N_p groups (similar to Split(N_p) after Step 1 in Figure 4.4). In other words, each group consists of the remainder values for the same time but different cycles.

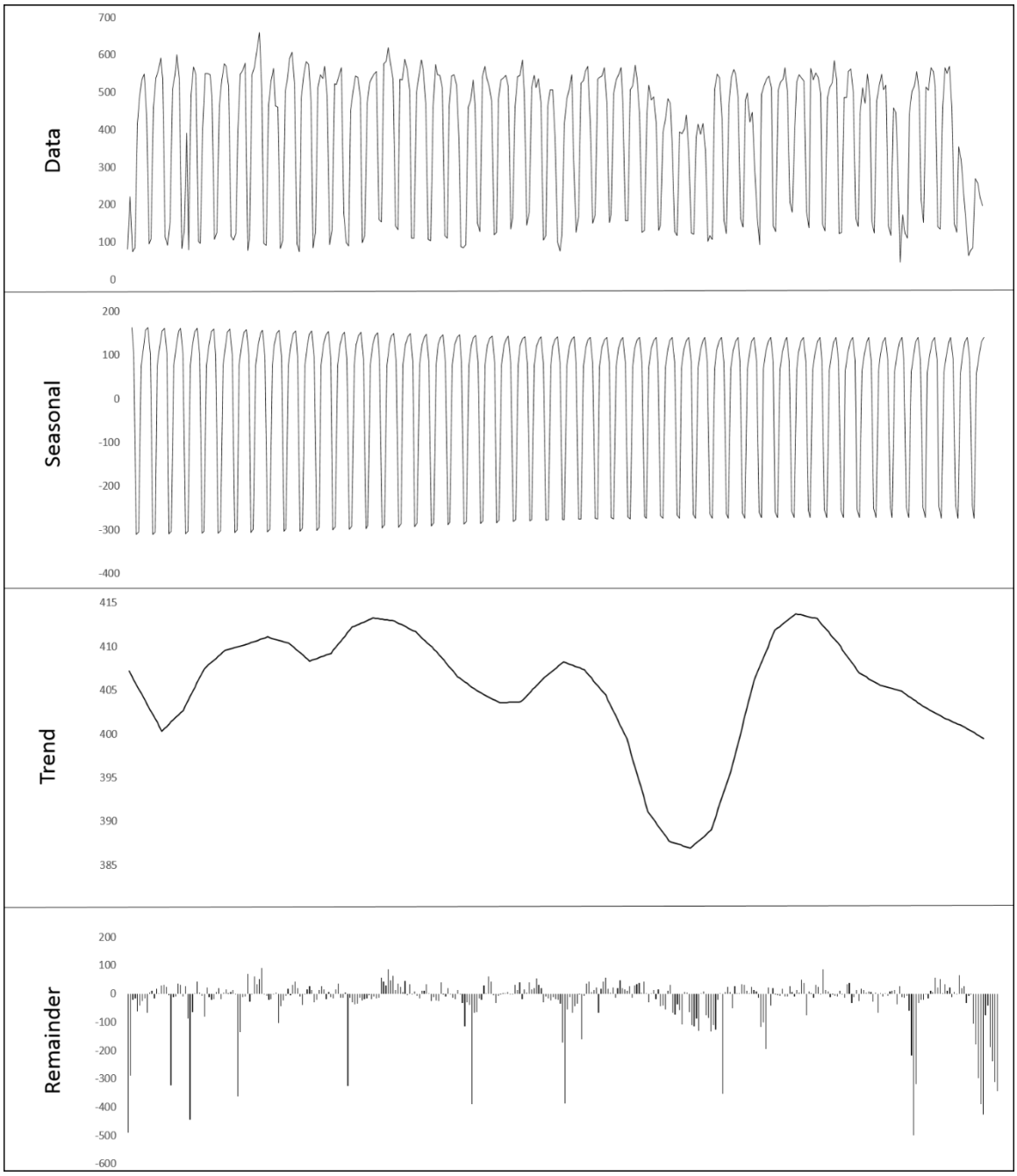

Figure 4.5 Three output components of the STL time series decomposition for the daily time series data of seven years at 200 West Street (Goldman Sachs Tower), with N_p =7, N_s=53, N_t = 11, N_o=20, and N_i=1. Only year 2015 is shown here due to limited space.

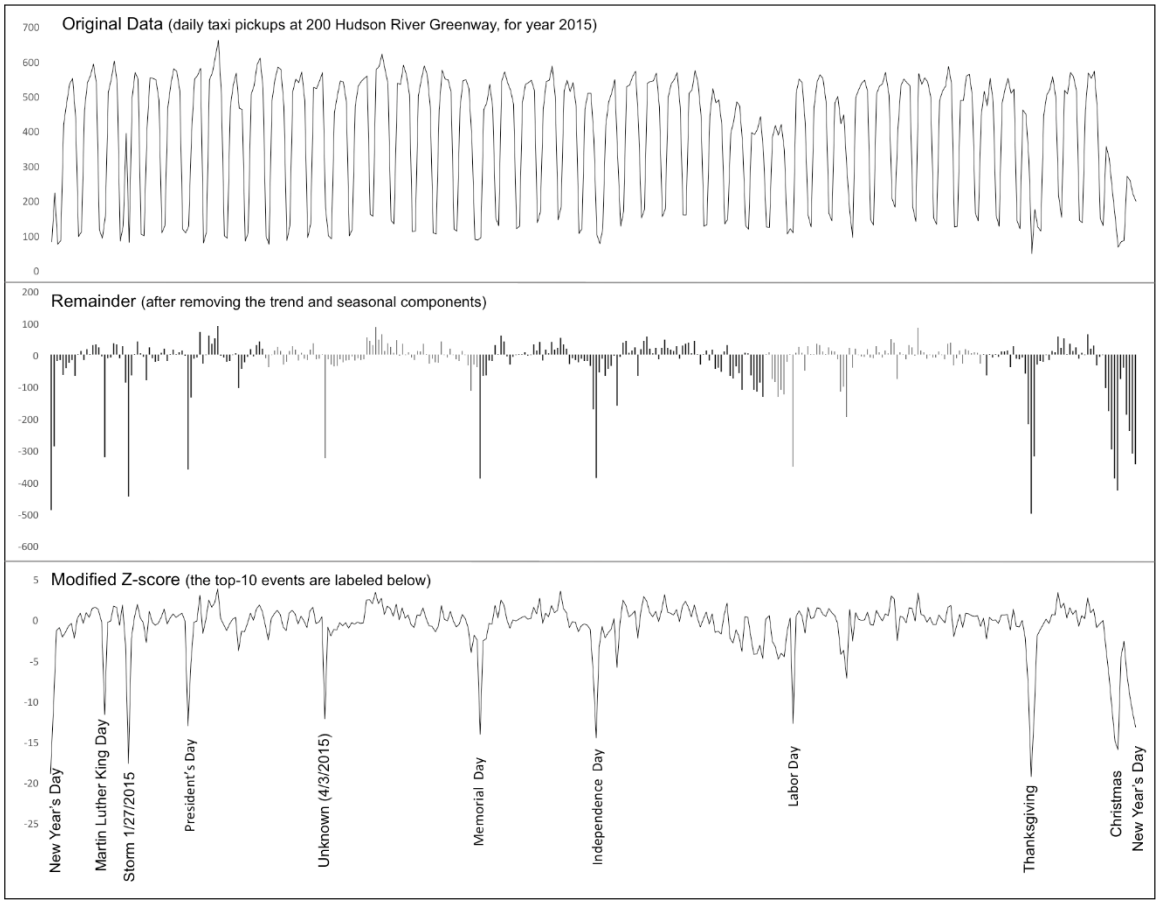

Figure 4.6 The modified z-score values and detected top events at 200 West Street (Goldman Sachs Tower) using the daily time series of taxi pickups for seven years. Only 2015 is shown in the plot due to limited space.

For example, if the input data is daily observational values and Np = 7 (i.e., weekly periodicity), the remainder values will be divided into seven groups, where each group has the remainder values for the same week day, e.g., Saturday. As such, an event is evaluated against those remainder values for the same weekdays.

We use two alternative methods to extract events from the remainder component. The first method is based on the Tukey's range test, with which outliers (events) in the remainder component are those values that are outside the range:$[Q_1 - k(Q_3 - Q_1), Q_3 +$

$k(Q_3 - Q_1)]$,, where Q_1 and Q_3 are the lower and upper quartiles of the group, and k usually is between 1.5 and 3. This test can identify events but does not provide a quantitative measurement of events. To more accurately quantify events, we also calculate a modified Z-score for each remainder value in the group. The modified Z-score (M_i) for a remainder value R_i in a group is computed as: $M_i = 0.6745*(R_i - \mu)/MAD$, where μ is the median of the group and MAD is the median of the absolute deviation to the median. Figure 4.6 shows the modified z-scores for the 2015 data as shown in Figure 4.5. There are 10 major events with $M_i > 10$ for the year, which not only covers all the major holidays but also includes unusual events such as the Blizzard of January 26–27, 2015 and another event on Friday, April 3.

4.5.3 Parameter configuration

The STL method has three important parameters: (1) N_p – the number of observations (or the length) of a seasonal cycle; (2) N_s – the size of the smoothing window for LOESS in step 2 (cycle-subseries smoothing); and (3) N_o – the number of iterations for the outer loop. Other parameters can be determined accordingly or using a default setting, such as N_i – the number of iterations of the inner loop, and N_t – the smoothing window size for LOESS in step 6 (trend smoothing).

Parameter setting is related to both the resolution of the time series (e.g., hourly, daily or weekly) and the periodicity cycle length (which can be determined with domain knowledge or by a data-driven method such as the Discrete Fourier Transform). For example, the taxi data can be transformed to hourly time series, daily time series or weekly time series. There are also a number of well-known periodicity cycles that exist in the data,

such as daily and weekly patterns. Parameters will be set according to the data resolution and the periodicity cycle to be captured. In the following subsections, we will discuss parameter configuration in more detail. To facilitate the discussion of parameter configuration and case studies in Section 4.6, we define three scenarios here:

1. Scenario 1: taxi data aggregated to hourly time series, assuming both daily and weekly periodicity;

2. Scenario 2: taxi data aggregated to daily time series, assuming weekly periodicity (results in Figure 4.5 and Figure 4.6 are from this scenario); and

3. Scenario 3: weekly time series with assumed annual periodicity.

4.5.3.1 Parameter N_p

Np is the number of observations (or the length) of a periodicity cycle, which can be either set based on domain knowledge or discovered with a periodicity test method such as the Discrete Fourier Transform (DFT). In our case study, for Scenario 1 (where the input data is hourly time series), we set Np = 168 since one week has 168 hours, which can capture both daily and weekly periodicity. Similarly, we set Np = 7 for Scenario 2 since the data is daily and one week has 7 days. For Scenario 3, Np = 52 for the weekly data with assumed annual periodicity.

4.5.3.2 Parameter N_t and N_s

STL is flexible to allow different levels of variation in the trend and seasonal components, which can be controlled by the parameters N_t (the smoothing window size for the trend component) and N_s (the smoothing window size for seasonal periodicity). The

choice of these two parameters in practice is based on the understanding of which variation in a time series should be considered seasonal periodicity and which should be in the trend component. For example, an "event" that lasts for a long time (e.g., a few months) may be viewed as trend, in which case N_t should be a set to a relatively small value so that the trend component captures more variation. With a larger N_t, the trend becomes smoother and more variation goes into the seasonal and remainder components.

Similarly, a larger value of Ns will produce a smoother seasonal component and capture less variation. For both Scenarios 1 and 2, the main seasonal periodicity is weekly and we set N_s= 53 to smooth the seasonal component with a one-year window (since there are 52 weeks in one year and Ns needs to be an odd number). Ns should not be less than 7 for the smoothing process to be meaningful. Once N_p and N_s are set, the value of N_t can be set according to the following inequality (Cleveland et al. 1990):

$$N_t \geq \frac{1.5N_p}{1 - 1.5N_s^{-1}}$$

In our case studies, for Scenario 1, N_p = 168, N_s = 53, and N_t = 259. For Scenario 2, N_p = 7, N_s = 53, and N_t = 11. For Scenario 3, N_p = 52, N_s = 7 (this is the smallest value possible for this parameter), and N_t = 99.

4.5.3.3 Parameter N_o

The number of iterations for the outer loop, N_o, is important for fine-tuning the robustness weights, which reduces the influence of noise and outliers on the estimation of the trend and seasonal components. No should be large enough to allow the estimation of the trend and for seasonal components to converge. We set N_o = 20 based on experiments with a convergence test. When set, the number of runs of the inner loop N_i = 1 since N_o =

20 is large enough (Cleveland et al. 1990). The outer loop iterations essentially are to make sure that the remainder component reliably captures outlier values.

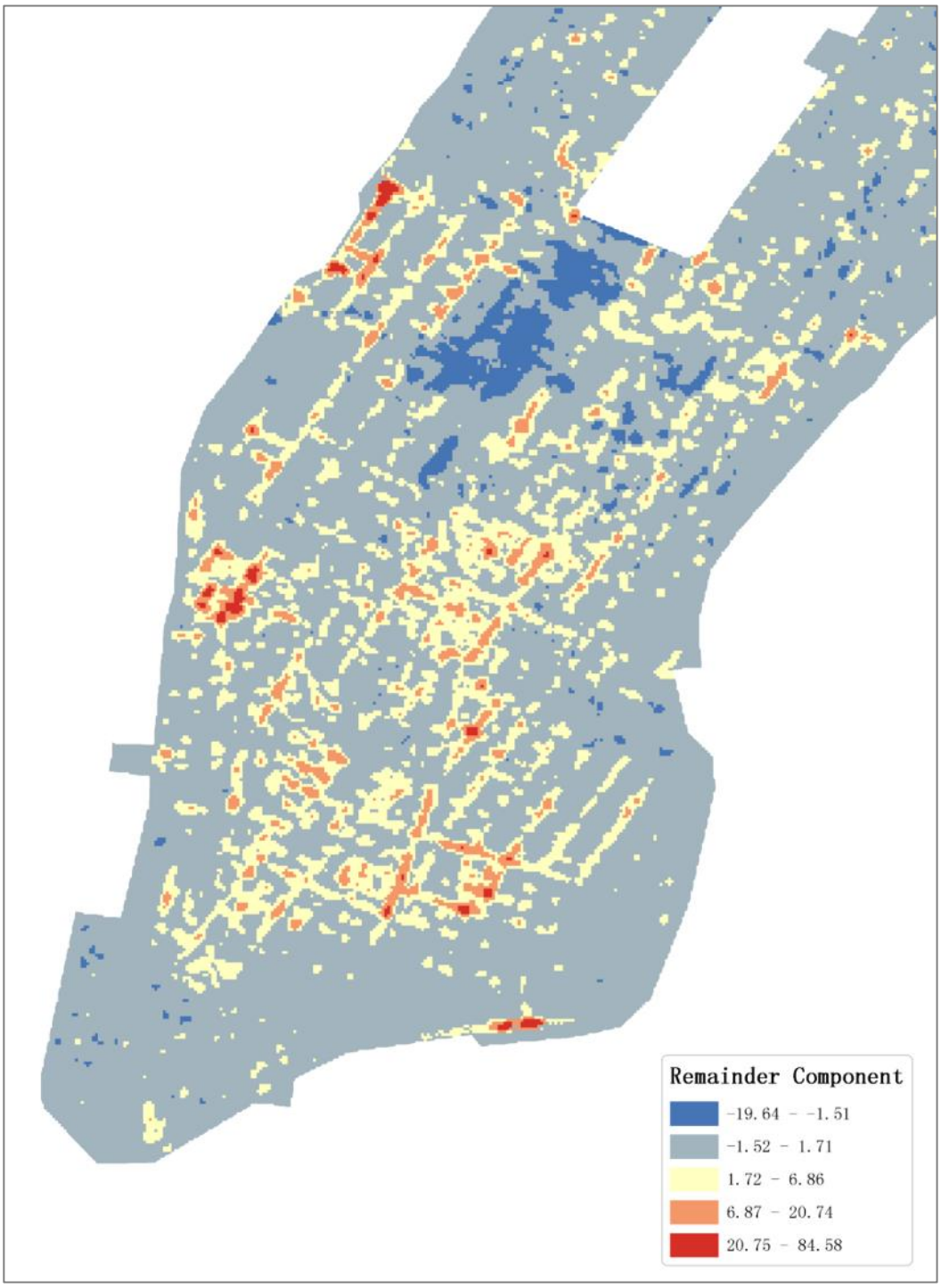

Figure 4.7 The remainder component of taxi arrivals at 11 pm on the New Year's Eve, 2013.

4.6 CASE STUDY AND RESULTS

To evaluate the effectiveness of our approach, we analyzed the big dataset of taxi trips as introduced in Section 4.4, which has over one billion (1,179,731,355) taxi OD trips in Manhattan (New York City) for seven years (2009—2015). On average, each day has about half a million taxi-trips with accurate location (about 5 m accuracy) and time (in second). We divide the study area into 20 by 20 m grids, altogether 137,046 grid cells.

For each grid cell, we find all taxi arrivals/departures within a 50 m buffer to the grid center and construct three different types of time series for the three scenarios (see Section 4.5), respectively. For each time series, our approach decomposes the data into three components: long-term trend, seasonal periodicity, and the remainder. While both the long-term trend and seasonal periodicity are interesting, in this research we focus on detecting events from the remainder component. Events can be either an unusually high concentration of taxi arrivals/pickups (called positive events) or an unusually low volume of arrivals/pickups (called negative events).

Our approach accurately detected a collection of interesting events at different spatial and temporal scales (with the three different scenarios), ranging from regional events such as festivals, hurricanes and snowstorms to local events such as exhibitions and football games. To help understand and evaluate the detected events, below we present a number of representative examples. Figure 4.7 shows an overview map of the remainder values (one for each grid) at 11 p.m. on 31 December, 2013. Based on these remainder values, events will be detected. With the map, we can identify the hot spots of taxi arrivals/drop-offs, i.e., locations with higher-than-expected arrivals (represented by red colors) and locations with unusually low volume of taxi arrivals (in blue) on New Year's Eve. It is

interesting to see that the area (in blue) to the south of the Central Park was receiving fewer-than-expected taxi arrivals at the time. We further checked other sources of information and found that the area was indeed under traffic control and thus of limited access for taxi cars (or any other vehicles) at that moment for New Year's Eve, due to the annual celebration at Time Square.

Figure 4.8 shows the locations with 300 or more extracted events in the hourly time series over the seven years. Among the 137,046 grids for the whole area, only 1,277 grids have a frequency of events greater than 300. Further examination of these locations, we find that most of them are the landmarks in NYC, such as Museums and Train Stations. However, we also find locations that are less "well-known", e.g., Pier 92/94, and the 287 Gallery, which often hold events and gatherings.

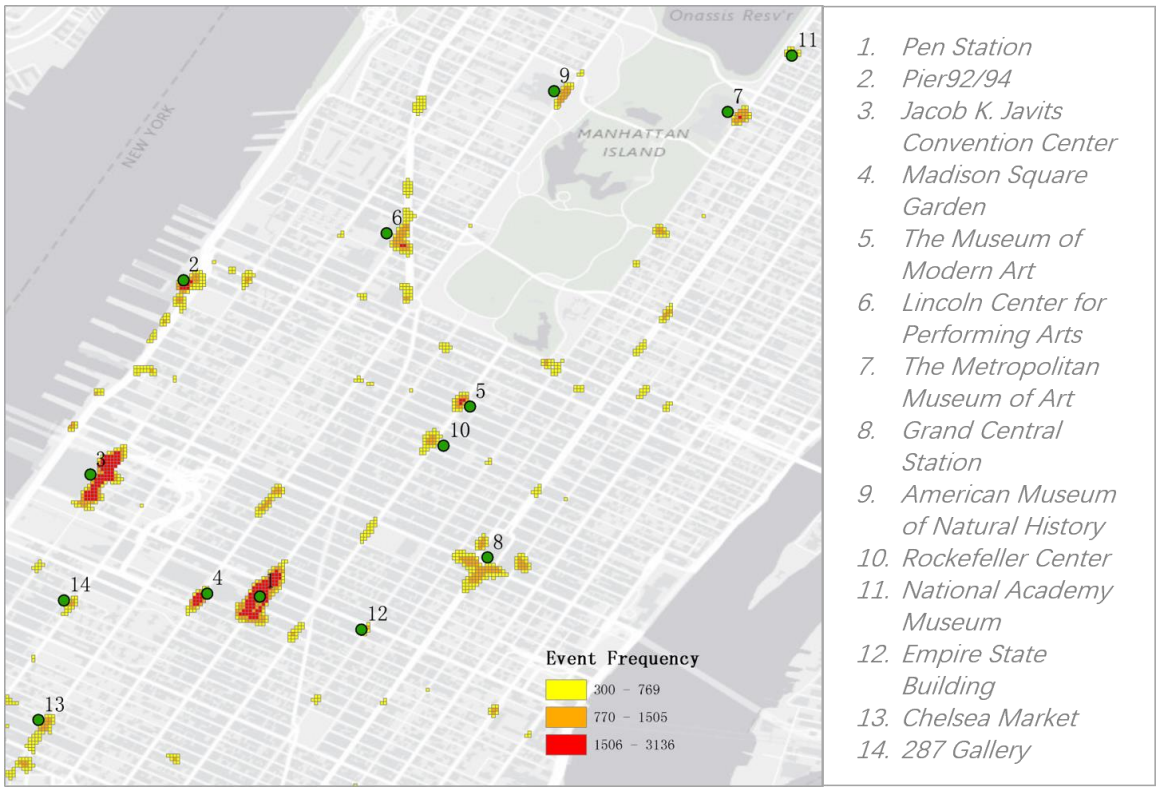

Figure 4.8 Locations with most events in seven years (2009 - 2015)

Further analyses of the characteristics and temporal distribution of the events for each location can help better understand the events and infer the land use type or function of the location. For example, Figure 4.9 shows the temporal distribution of the events that occurred at Little Italy (near Low Manhattan). Please note, these week-long events were extracted from a weekly time series, and it classified trends in hourly and daily time series decomposition. In the chart, the blue color represents negative events (when there were fewer taxi arrivals than normal), the red color represents positive events, and the size of the circle represents the significance of the event. We can immediately see that every year in September there is a big event in Little Italy, from which we later learned that there is indeed a Feast of San Gennaro at Little Italy in mid-September every year, when the area will be blocked off and therefore offers no taxi access for arrivals.

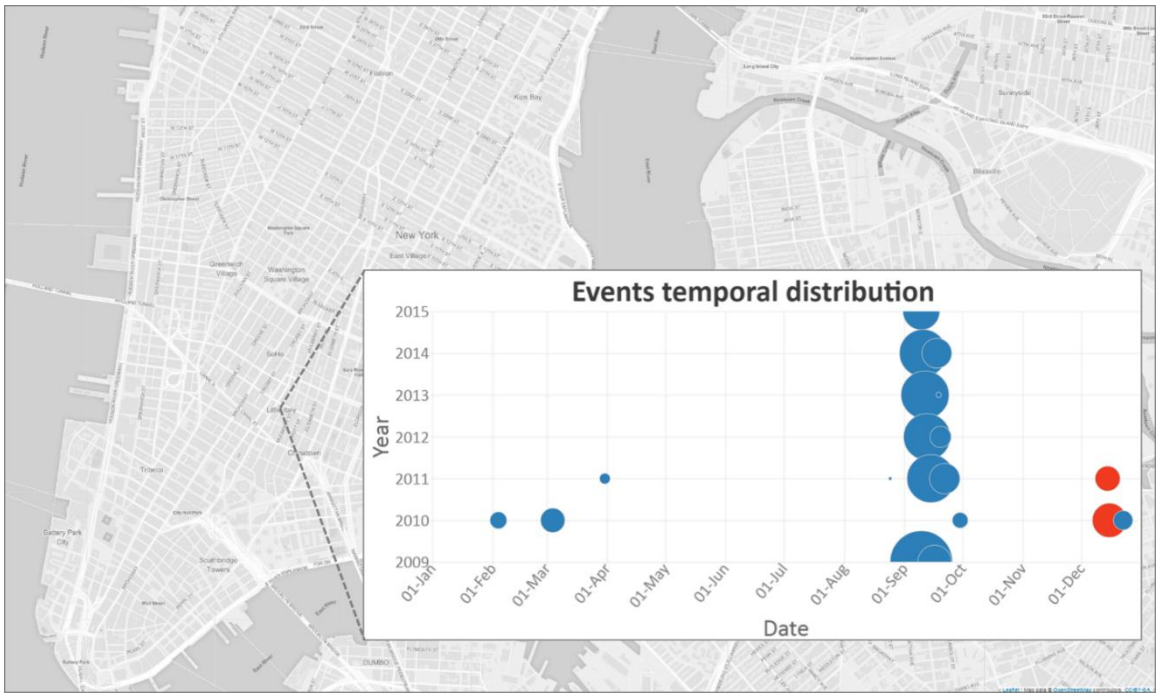

Figure 4.9 Temporal distribution of the events happened at the Little Italy. There is Annual Feast of San Gennaro at Little Italy every middle September. The blue color represents events which have fewer taxi arrivals than normal, and red represents the opposite. The size of the circle represents the significance of the events.

4.7 CONCLUSIONS AND FUTURE WORK

This article presents a new approach to urban event detection and analysis with big data of taxi trips. Built upon the STL time series decomposition method, our approach decomposes a time series (one for each location) into three components: the trend, season periodicity, and the remainder. Events are then detected from the remainder component alone, separated from the trend and periodicity. As such, an event is defined as a remainder value that is significantly different from its expected value based on the discovered trend and periodicity. Previous event detection approaches either assume a random distribution (i.e., does not consider trend or periodicity) or only take into account periodicity (but do not consider the long-term trend). With the case study result, as shown in Sections 4.5 and 4.6, we have shown that the approach is effective in detecting events and revealing urban dynamics with both a holistic understanding and location-specific interpretations.

While this article primarily focuses on the remainder component and event detection, the discovered trend, periodicity and events can be used for further analysis of urban patterns such as land use functions, dynamics and changes. The approach can also be used for analyzing other mobility data such as mobile phone data and social media activities. Future work may also examine the origin-destination relationship and analyze the geographic impacts of specific events. In this article, the spatial resolution (i.e., grid partitioning) is fixed. It will be very interesting to extend the approach further to automatically detect events of different geographic sizes and temporal lengths. While the data being used in this study is not a real-time data feed, it is possible to extend the method to enable real-time event detection based on immediate past or historical records.

REFERENCES

Adrienko, N. & G. Adrienko (2011) Spatial generalization and aggregation of massive movement data. *IEEE Transactions on visualization and computer graphics,* 17**,** 205-219.

Ahas, R., A. Aasa, Ü. Mark, T. Pae & A. Kull (2007) Seasonal tourism spaces in Estonia: Case study with mobile positioning data. *Tourism Management,* 28**,** 898-910.

Ahas, R., S. Silm, O. Järv, E. Saluveer & M. Tiru (2010) Using mobile positioning data to model locations meaningful to users of mobile phones. *Journal of Urban Technology,* 17**,** 3-27.

Alper, B., B. Bach, N. Henry Riche, T. Isenberg & J.-D. Fekete. 2013. Weighted graph comparison techniques for brain connectivity analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 483-492. ACM.

Andrienko, G. & N. Andrienko (2002) A general framework for using aggregation in visual exploration of movement data. *The Cartographic Journal,* 47**,** 22-40.

Andrienko, G., N. Andrienko, J. Dykes, S. I. Fabrikant & M. Wachowicz. 2008. Geovisualization of dynamics, movement and change: key issues and developing approaches in visualization research. SAGE Publications Sage UK: London, England.

Andrienko, G., N. Andrienko, G. Fuchs & J. Wood (2016) Revealing Patterns and Trends of Mass Mobility through Spatial and Temporal Abstraction of Origin-Destination Movement Data. *IEEE transactions on visualization and computer graphics*.

Andrienko, N. & G. Andrienko (2011) Spatial Generalization and Aggregation of Massive Movement Data. *IEEE Transactions on Visualization and Computer Graphics,* 17**,** 205 - 219.

Bianco, A. M., M. Garcia Ben, E. Martinez & V. J. Yohai (2001) Outlier detection in regression models with arima errors using robust estimates. *Journal of Forecasting,* 20**,** 565-579.

Boyandin, I., E. Bertini, P. Bak & D. Lalanne. 2011. Flowstrates: An Approach for Visual Exploration of Temporal Origin-Destination Data. In *Computer Graphics Forum*, 971-980. Wiley Online Library.

Breunig, M. M., H.-P. Kriegel, R. T. Ng & J. Sander. 2000. LOF: identifying density-based local outliers. In *ACM sigmod record*, 93-104. ACM.

Buchin, K., B. Speckmann & K. Verbeek (2011) Flow map layout via spiral trees. *IEEE transactions on visualization and computer graphics,* 17**,** 2536-2544.

Calabrese, F., V. Blondel & L. Ferrari (2014) Urban Sensing Using Mobile Phone Network Data: A Survey of Research. *ACM Computing Surveys: the survey and tutorial journal of the ACM*, 47.

Calabrese, F., M. Colonna, P. Lovisolo, D. Parata & C. Ratti (2011) Real-time urban monitoring using cell phones: A case study in Rome. *IEEE Transactions on Intelligent Transportation Systems,* 12**,** 141-151.

Calabrese, F., F. C. Pereira, G. Di Lorenzo, L. Liu & C. Ratti. 2010. The geography of taste: analyzing cell-phone mobility and social events. In *International Conference on Pervasive Computing*, 22-37. Springer.

Candia, J., M. C. González, P. Wang, T. Schoenharl, G. Madey & A.-L. Barabási (2008) Uncovering individual and collective human dynamics from mobile phone records. *Journal of Physics A: Mathematical and Theoretical*, 41**,** 224015.

Chae, J., D. Thom, H. Bosch, Y. Jang, R. Maciejewski, D. S. Ebert & T. Ertl. 2012. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, 143-152. IEEE.

Chen, L. & A. Roy. 2009. Event detection from flickr data through wavelet-based spatial analysis. In *Proceedings of the 18th ACM conference on Information and knowledge management*, 523-532. ACM.

Cleveland, R. B., W. S. Cleveland, J. E. McRae & I. Terpenning (1990) STL: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, 6**,** 3-73.

Cleveland, W. S. (1979) Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association*, 74**,** 829-836.

Cui, W., H. Zhou, H. Qu, P. C. Wong & X. Li (2008) Geometry-based edge clustering for graph visualization. *IEEE Transactions on Visualization and Computer Graphics*, 14**,** 1277-1284.

Deville, P., C. Linard, S. Martin, M. Gilbert, F. R. Stevens, A. E. Gaughan, V. D. Blondel & A. J. Tatem (2014) Dynamic population mapping using mobile phone data. *Proceedings of the National Academy of Sciences,* 111**,** 15888-15893.

Dong, X., D. Mavroeidis, F. Calabrese & P. Frossard (2015a) Multiscale event detection in social media. *Data Mining and Knowledge Discovery,* 29**,** 1374-1405.

Dong, Y., F. Pinelli, Y. Gkoufas, Z. Nabi, F. Calabrese & N. V. Chawla. 2015b. Inferring unusual crowd events from mobile phone call detail records. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 474-492. Springer.

Dwyer, T., B. Lee, D. Fisher, K. I. Quinn, P. Isenberg, G. Robertson & C. North (2009) A comparison of user-generated and automatic graph layouts. *IEEE transactions on visualization and computer graphics,* 15.

Ester, M., H.-P. Kriegel, J. Sander & X. Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, 226-231.

Ferreira, N., J. Poco, H. T. Vo, J. Freire & C. T. Silva (2013) Visual exploration of big spatio-temporal urban data: A study of new york city taxi trips. *IEEE Transactions on Visualization and Computer Graphics,* 19**,** 2149-2158.

Fryxell, J. M., M. Hazell, L. Börger, B. D. Dalziel, D. T. Haydon, J. M. Morales, T. McIntosh & R. C. Rosatte (2008) Multiple movement modes by large herbivores at multiple spatiotemporal scales. *Proceedings of the National Academy of Sciences of the United States of America*.

Ghoniem, M., J.-D. Fekete & P. Castagliola. 2004. A comparison of the readability of graphs using node-link and matrix-based representations. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, 17-24. Ieee.

Guo, D. (2009) Flow mapping and multivariate visualization of large spatial interaction data. *IEEE Transactions on Visualization and Computer Graphics,* 15.

Guo, D. & M. Gahegan (2006) Spatial ordering and encoding for geographic data mining and visualization. *Journal of Intelligent Information Systems,* 27**,** 243-266.

Guo, D. & X. Zhu (2014) Origin-destination flow data smoothing and mapping. *IEEE Transactions on Visualization and Computer Graphics,* 20**,** 2043-2052.

Guo, D., X. Zhu, H. Jin, P. Gao & C. Andris (2012) Discovering Spatial Patterns in Origin-Destination Mobility Data. *Transactions in GIS,* 16**,** 411-429.

Gupta, M., J. Gao, C. Aggarwal & J. Han (2014) Outlier detection for temporal data. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 5**,** 1-129.

Hafen, R. P., D. E. Anderson, W. S. Cleveland, R. Maciejewski, D. S. Ebert, A. Abusalah, M. Yakout, M. Ouzzani & S. J. Grannis (2009) Syndromic surveillance: STL for modeling, visualizing, and monitoring disease counts. *BMC Medical Informatics and Decision Making,* 9**,** 21.

Holten, D. & J. J. van Wijk (2009) Force-Directed Edge Bundling for Graph Visualization. *Computer Graphics Forum,* 28**, 983–990**983–990.

Jenny, B., D. M. Stephen, I. Muehlenhaus, B. E. Marston, R. Sharma, E. Zhang & H. Jenny (2016) Design principles for origin-destination flow maps. *Cartography and Geographic Information Science***,** 1-15.

--- (2017) Force-directed layout of origin-destination flow maps. *International Journal of Geographical Information Science,* 1-20.

Jones, M. C., J. S. Marron & S. J. Sheather (1996) A brief survey of bandwidth selection for density estimation. *Journal of the American Statistical Association,* 91**,** 401-407.

Koylu, C. & D. Guo (2016) Design and evaluation of line symbolizations for origin–destination flow maps. *Information Visualization***,** 1473871616681375.

Kraak, M.-J. 2003. The space-time cube revisited from a geovisualization perspective. In *Proc. 21st International Cartographic Conference*, 1988-1996.

Kraak, M.-J. & A. Koussoulakou. 2005. A visualization environment for the space-time-cube. In *Developments in Spatial Data Handling*, 189-200. Springer.

Kulldorff, M. (1997) A spatial scan statistic. *Communications in Statistics-Theory and methods,* 26**,** 1481-1496.

Kut, A. & D. Birant (2006) Spatio-temporal outlier detection in large databases. *CIT. Journal of computing and information technology,* 14**,** 291-297.

Laube, P. & R. S. Purves (2011) How fast is a cow? Cross‐Scale Analysis of Movement Data. *Transactions in GIS*, 15**,** 401-418.

Liu, Y., X. Liu, S. Gao, L. Gong, C. Kang, Y. Zhi, G. Chi & L. Shi (2015) Social sensing: A new approach to understanding our socioeconomic environments. *Annals of the Association of American Geographers,* 105**,** 512-530.

Lu, C.-T. & L. R. Liang. 2004. Wavelet fuzzy classification for detecting and tracking region outliers in meteorological data. In *Proceedings of the 12th annual ACM international workshop on Geographic information systems*, 258-265. ACM.

Lu, J., G. J. Carbone & P. Gao (2017) Detrending crop yield data for spatial visualization of drought impacts in the United States, 1895–2014. *Agricultural and Forest Meteorology,* 237**,** 196-208.

Nathan, R., W. M. Getz, E. Revilla, M. Holyoak, R. Kadmon, D. Saltz & P. E. Smouse (2008) A movement ecology paradigm for unifying organismal movement research. *Proceedings of the National Academy of Sciences,* 105**,** 19052-19059.

Pan, G., G. Qi, Z. Wu, D. Zhang & S. Li (2013) Land-use classification using taxi GPS traces. *Intelligent Transportation Systems, IEEE Transactions on,* 14**,** 113-123.

Phan, D., L. Xiao, R. Yeh & P. Hanrahan. 2005. Flow map layout. In *Information Visualization, 2005. INFOVIS 2005. IEEE Symposium on,* 219-224. IEEE.

Pincombe, B. (2005) Anomaly detection in time series of graphs using arma processes. *Asor Bulletin,* 24**,** 2.

Purchase, H. C., J. Hamer, M. Nöllenburg & S. G. Kobourov. 2012. On the usability of Lombardi graph drawings. In *International Symposium on Graph Drawing,* 451-462. Springer.

Rae, A. (2009) From spatial interaction data to spatial interaction information? Geovisualisation and spatial structures of migration from the 2001 UK census. *Computers Environment and Urban Systems,* 33.

--- (2011) Flow-data analysis with geographical information systems: a visual approach. *Environment and Planning B-Planning & Design,* 38.

Rattenbury, T., N. Good & M. Naaman. 2007. Towards automatic extraction of event and place semantics from flickr tags. In *Proceedings of the 30th annual international*

ACM SIGIR conference on Research and development in information retrieval, 103-110. ACM.

Ratti, C., D. Frenchman, R. M. Pulselli & S. Williams (2006) Mobile landscapes: using location data from cell phones for urban analysis. *Environment and Planning B: Planning and Design,* 33**,** 727-748.

Sagl, G., B. Resch, B. Hawelka & E. Beinat. 2012. From social sensor data to collective human behaviour patterns: Analysing and visualising spatio-temporal dynamics in urban environments. In *In Proceedings of the GI-Forum 2012: Geovisualization, Society and Learning*. Salzburg: Wichmann.

Sakaki, T., M. Okazaki & Y. Matsuo. 2010. Earthquake shakes Twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, 851-860. ACM.

--- (2013) Tweet analysis for real-time event detection and earthquake reporting system development. *IEEE Transactions on Knowledge and Data Engineering,* 25**,** 919-931.

Scheepens, R., N. Willems, H. Van de Wetering, G. Andrienko, N. Andrienko & J. J. Van Wijk (2011a) Composite density maps for multivariate trajectories. *IEEE Transactions on Visualization and Computer Graphics,* 17**,** 2518-2527.

Scheepens, R., N. Willems, H. van de Wetering & J. van Wijk (2012) Interactive density maps for moving objects. *IEEE Computer Graphics and Applications,* 32**,** 56-66.

Scheepens, R., N. Willems, H. van de Wetering & J. J. van Wijk. 2011b. Interactive visualization of multivariate trajectory data with density maps. In *Pacific Visualization Symposium (PacificVis), 2011 IEEE*, 147-154. IEEE.

Scholz, R. W. & Y. Lu (2014) Detection of dynamic activity patterns at a collective level from large-volume trajectory data. *International Journal of Geographical Information Science,* 28**,** 946-963.

Sheather, S. J. & M. C. Jones (1991) A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society. Series B (Methodological)***,** 683-690.

Silverman, B. W. 1986. *Density estimation for statistics and data analysis*. Chapman & Hall/CRC.

Soleymani, A., J. Cachat, K. Robinson, S. Dodge, A. Kalueff & R. Weibel (2015) Integrating cross-scale analysis in the spatial and temporal domains for classification of behavioral movement. *Journal of Spatial Information Science***,** 1-25.

Tao, R. & J. C. Thill (2016) Spatial cluster detection in spatial flow data. *Geographical Analysis,* 48**,** 355-372.

Thomas, J. J. & K. Cook (2006) A visual analytics agenda. *Computer Graphics and Applications, IEEE,* 26**,** 10-13.

Tobler, W. R. (1981) A model of geographical movement. *Geographical Analysis,* 13**,** 1-20.

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

Traag, V. A., A. Browet, F. Calabrese & F. Morlot. 2011. Social event detection in massive mobile phone data using probabilistic location inference. In *Privacy, security, risk and trust (PASSAT) and 2011 IEEE Third inernational conference on social*

computing (SocialCom), 2011 IEEE Third International Conference on, 625-628. IEEE.

Verbeek, K., K. Buchin & B. Speckmann (2011) Flow map layout via spiral trees. *IEEE transactions on visualization and computer graphics,* 17**,** 2536-2544.

Verbesselt, J., R. Hyndman, G. Newnham & D. Culvenor (2010a) Detecting trend and seasonal changes in satellite image time series. *Remote sensing of Environment,* 114**,** 106-115.

Verbesselt, J., R. Hyndman, A. Zeileis & D. Culvenor (2010b) Phenological change detection while accounting for abrupt and gradual trends in satellite image time series. *Remote Sensing of Environment,* 114**,** 2970-2980.

Ware, C., J. G. Kelley & D. Pilar (2014) Improving the display of wind patterns and ocean currents. *Bulletin of the American Meteorological Society,* 95**,** 1573-1581.

Willems, N., R. Scheepens, H. van de Wetering & J. J. van Wijk. 2013. Visualization of vessel traffic. In *Situation Awareness with Systems of Systems,* 73-87. Springer.

Wood, J., J. Dykes & A. Slingsby (2010) Visualisation of origins, destinations and flows with OD maps. *The Cartographic Journal,* 47**,** 117-129.

Wu, E., W. Liu & S. Chawla. 2010. Spatio-temporal outlier detection in precipitation data. In *Knowledge discovery from sensor data,* 115-133. Springer.

Xu, K., C. Rooney, P. Passmore, D.-H. Ham & P. H. Nguyen (2012) A user study on curved edges in graph visualization. *IEEE Transactions on Visualization and Computer Graphics,* 18**,** 2449-2456.

Yan, J. & J.-C. Thill (2009) Visual data mining in spatial interaction analysis with self-organizing maps. *Environment and Planning B,* 36**,** 466-486.

Zhang, W., G. Qi, G. Pan, H. Lu, S. Li & Z. Wu (2015) City-scale social event detection and evaluation with taxi traces. *ACM Transactions on Intelligent Systems and Technology (TIST),* 6**,** 40.

Zhu, X. & D. Guo (2014) Mapping large spatial flow data with hierarchical clustering. *Transactions in GIS,* 18**,** 421-435.