Exploring OD Patterns of Interested Region based on Taxi Trajectories

Min Lu[1,2] Zuchao Wang[1] Jie Liang[1] * Xiaoru Yuan[1,2] †

1) Key Laboratory of Machine Perception (Ministry of Education), and School of EECS, Peking University
2) Center for Computational Science and Engineering, Peking University

Abstract—Different regions in the city probably have different Origin-Destination (OD) patterns. Given a region, related OD patterns potentially reveals its surrounding traffic context and social functions. In this work, we present a visual analytics system to explore such OD patterns of interested region based on taxi trajectories. The system integrates interactive trajectory filtering with OD patterns visual exploration. Trajectories related to interested region are selected by a suite of graphical filtering tool, from which OD clusters are detected automatically. OD traffic patterns are explored from two levels: overview of spatial and statistical distribution and detail exploration on dynamic OD patterns, including dynamic traffic volume and travel time. By applying to real taxi trajectory data set, we demonstrate the effectiveness of our system by several case studies.

Index Terms—OD Pattern, Trajectory, Filter

1 Introduction

Understanding the movement of human beings and vehicles in large cities is crucial in transportation field. Such movement, known as the Origin-Destination (OD) pattern, reveals urban traffic context including the hot O/D places, regular traffic flows among regions. For example, regular commuting patterns can be extracted by studying the flow volume and travel time of traffic flow [21]. On the other hand, the dynamic changes of OD patterns potentially implies regions' social functions, for example land-use patterns can be inferred by studying the dynamics of traffic flow related to regions [20].

OD patterns in urban can be indicated from different types of movement data. Taxis travel around the city in a considerable spatial and temporal scale. They are viewed as the representative of urban traffic. In visualization and data mining communities, many traffic related researches are conducted based on taxi trajectory data. For example, by studying taxis' traffic speed, irregular traffic congestion patterns can be identified [15, 27]. Extracting from trajectories passing through certain routes, either single route's traffic condition [18] or diversity of multiple routes [14] between regions are studied. Hence, considering taxi trajectory as a typical urban movement data, we use taxi trajectories to study urban OD patterns in this work.

Different from work to study the global traffic flow all over the city [3], we focus on local OD patterns analysis related to region of interest, which may be dimmed in global analysis. Given the region of interest, our goal is to extract its surrounding OD regions and dynamic OD and explore their OD patterns. Specifically, we target at exploring OD patterns from two levels:

- **Overview of OD regions:** to give an overview of spatial distribution (e.g. where the traffic flows mainly come/go) and traffic related statistical distribution of OD patterns.
- **Detail exploration on OD patterns:** to explore travel flow volume, travel time cost of OD pair in detail, e.g. how the traffic volume and traffic related measurement of traffic flow between OD pair changes along time.

*e-mail: {lumin.vis, zuchao.wang, christy.jie}@gmail.com
†e-mail: xiaoru.yuan@pku.edu.cn

Manuscript received 31 Mar. 2014; accepted 1 Aug. 2014; date of publication xx xxx 2014; date of current version xx xxx 2014. For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

According to the above two tasks, we extend the idea from OD-Wheel [17] which only explores the OD clusters starting from/ending at a central region, to a visual analytics system which supports exploration on OD regions detected from filtered trajectories. The system integrates: (1) an intuitive visual query interface to define interest region and filter taxi trajectories; (2) an automatic clustering algorithm to extract OD clusters from taxi trajectories and detect corresponding regions; (3) spatial and statistical overview of OD clusters; (4) an adapted OD-Wheel to describe, compare and correlate the dynamic OD patterns in detail.

Following a review of the related work in Section 2, we give an overview of data, tasks and system in Section 3. In Section 4, we describe filters designed in this work. In Section 5, we explain the visual design on OD patterns exploration in detail. In Section 7, we introduce implementation details. We demonstrate three cases to show the effectiveness of our system in Section 8. Last comes the discussion in Section 9 and conclusion in Section 10.

2 Related Work

Our work is related to trajectory filtering, OD visualization and temporal data visualization. We discuss existing works in those areas below.

2.1 Trajectory Filtering

Trajectory filtering techniques aim at extracting a subset of trajectories satisfying specific conditions. By driving a spatial geometric query object (such as a circle), filtering result is dynamic updated. Many interactive query methods have been designed. MagicLens [8] is generic lens used in 2D spatial space. Lens with different functions embeds different visualization of selected area, such as a detailed lens which magnifies the region or road network lens which shows the road network of underlying area. For example, Kr*ü*ger et al.'s TrajectoryLenses [13] design circular filters in spatial filtering. Set operations are used to combine multiple circular filters. Ferreira et al. [7] also develop a system, but for taxi OD data. They allow users to define origin and destination by sketching polygons. Then OD queries are executed and the results are visualized in multiple views. Such methods can be directly used for trajectory data. In addition to fully visual queries, filtering can also be performed via spatial visual query language. The query command can be interactively composed by predefined icons which represent either the spatial objects or operators [4].

In our work, we develop graphical fitlering tool to help user define interested region and get related taxi trajectories. This is achieved with a technique similar to TrajectoryLens. However, our system supports more complex settings on geometric constraints.

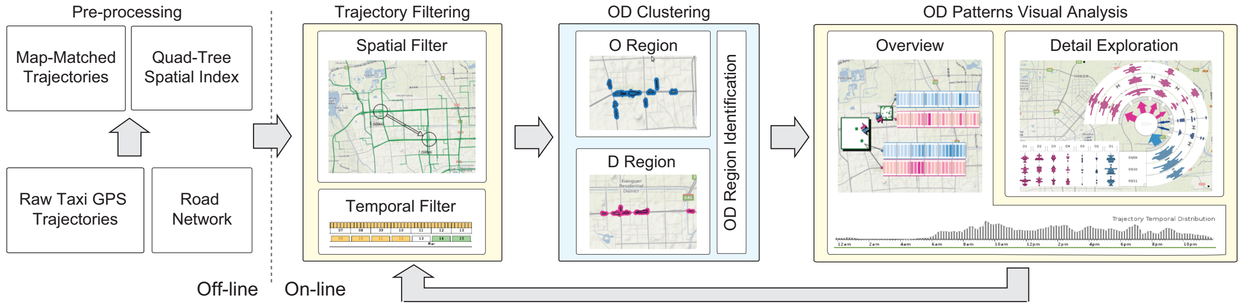

Fig. 1. System Pipeline: pipeline composes of four part: off-line trajectory preprocessing, interactive trajectory filter, automatic OD clusters extraction and temporal dynamic visualization.

2.2 OD Visualization

OD data can either be collected from discrete movement data [28] or aggregated from continuous trajectory data [3]. Such OD data can be visualized with a set of techniques, including flow map, OD matrix and OD map. Flow map [25] shows the origins and destinations as nodes on a geographic map. The directed links connecting the origin and destination nodes represent the traffic flows between the pair of OD. It is a special type of node-link diagram. Flow map is very intuitive, but it suffers from serious visual clutter. Possible solutions to alleviate such visual clutter include edge-filtering [23], edge-bundling [12], node hierarchy [10] or allowing only one origin or one destination [22]. OD matrix visualize the OD data in an abstract space. It represents origins and destinations as rows and columns of a matrix [2]. Each cell encodes a OD, whose flow magnitude is represented as color of the cell. Due to lack of spatial information, OD matrix are usually used in conjunction with a geographic map. OD map [30] makes a balance between clutter free and spatial information preservation. It partitions a geographic region to subregions with a 2D grid and define ODs between the subregions.

In our work, we visualize OD data from two views respectively. In spatial view, we focus on the spatial overview of ODs. In abstract view, the dynamics of traffic flow volume and travel time of ODs are explored within a hybrid linear-circular design.

2.3 Temporal Data Visualization

Temporal data visualization is one of common problems in visualization field. Animation is one of the most intuitive temporal data visualization techniques. For example, Whisper [6] visualizes the information diffusion in social media with animated pathways between origin tweet and retweet. However, it is widely believed that animation does not work well for analysis tasks [24]. Therefore, many static methods are developed.

If the data have some known periodicity, then calender view [26] and spiral view [29] can be good options. However, for general temporal data, timeline is the most popular methods. In a timeline, the x-axis represents time, and y-axis represents an attribute that changes with time. To compare different temporal data, each data can be visualized by a timeline. Then those timelines can be juxtaposed [19] or superimposed [11] for visual comparison. In case of juxtaposition, similar timelines can be put close to each other [16]. On the other hand, temporal data analysis can benefit from rich user interactions. Zhao et al. design KronoMiner [31] based on a circular layout of timelines. KronoMiner supports clicking based time range selection and dragging based data comparison.

Temporal data visualization techniques have been used to analyze OD data. In Ferreira et al.'s taxi trip exploration work [7], once users select multiple ODs and execute the query, the dynamics of trip number on each OD will be shown in a timeline. In Boyandin et al.'s Flowstrates [5], they analyze the refugeee flow with a specially designed three part interface. Two maps are used for origin and destination selection, while a heatmap is embedded between them to compare the flow volume between different ODs. Here the heatmap is in fact a juxtaposition of timelines.

In our work, we use a circular design for flow volume comparison, as that in KronoMiner. In addition, we also embed a linear timeline component for travel time comparison, as the heatmap. However, our analysis tasks and supported interactions are different from above works.

3 Overview

In this section, we first introduce the data concept used in the work. Then the tasks are clarified and an overview of our system is given.

3.1 Data

To facilitate the discussion, we list down some common terminologies in this work as illustrated in Figure 2:

- *Trajectory* is a list of positions in temporal order, recording the movement.
- *Origin/Destination (O/D)* refers to the beginning/ending position of the movement.
- *Origin/Destination Region* refers the region where lots of movements start or end, i.e. the clusters of *Origins* or *Destinations*.
- *OD Region* is the region where it is *Origin Region* and also the *Destination Region*.
- *Region of Interest* is generalized as interest region where *trajectories* travelling through in this work. It can either be a specific area, or a compound one of several areas which are combined in certain logic operations.

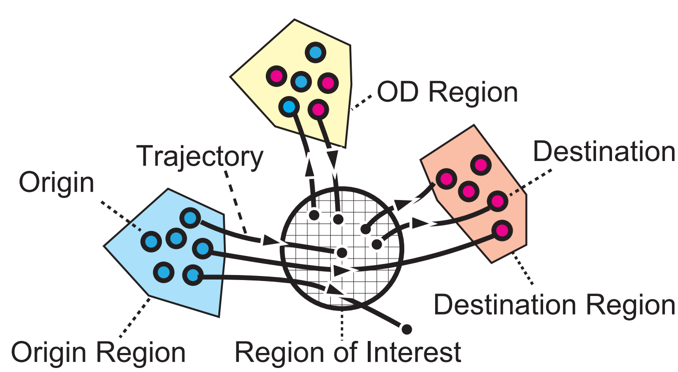

Fig. 2. Illustration of Traffic-related Concepts

Category	Attribute	Type	Formalization	Description (filtered out)
Spatial	Location	Origin	traj.origin ∈ region_defined	trajectories origining from defined region
		Destination	traj.destination ∈ region_defined	trajectories destinating for defined region
		Origin/Destination	traj.origin ∈ region_defined ∪ traj.destination ∈ region_defined	trajectories origining from or destinating for defined region
		Passing	traj.loctaion ∈ region_defined ∩ traj.origin ∉ region_defined ∩ traj.destination ∉ region_defined	trajectories passing through defined region
		Inclusion	traj.location ∈ region_defined	trajectories intersecting with defined region
		Exclusion	traj.location ∉ region_defined	trajectories not intersecting with defined region
	Direction	Direction	traj.direction = direction_defined	trajectories following defined direction
Temporal	Date	Date Range	traj.date ≥ begindate_defined ∩ traj.date ≤ enddate_defined	trajectories travelling in defined date range
	Time	Time Range	traj.time ≥ beigntime_defined ∩ traj.time ≤ endtime_defined	trajectories travelling in defined time range

Table 1. Filter Model: spatial and temporal filtering are supported, which combine with each other in intersection set operations.

3.2 Task

Starting from the two tasks introduced in Section 1, we specify the tasks supported by our system further, as following:

- **Extraction of O/D clusters related to region of interest (T1)**: to customize the region of interest and then extract the O/D, ODs regions based on the trajectories travelling through.
- **Spatial overview of O/D, OD regions (T2)**: to give an overview of how the regions distribute in spatial space.
- **Statistical overview of O/D, OD regions (T3)**: to show the statistical overview of regions.
- **Traffic volume comparison among regions (T4)**: to compare traffic volume among OD clusters.
- **Detail exploration OD pattern dynamic (T5)**: to explore traffic flow's dynamic measurements in detail, including travel time cost etc.

3.3 System Overview

To support exploring OD patterns of interested region, the system composes of four major parts: data preprocessing, trajectory filtering, OD clustering and visual analysis on OD patterns. Its pipeline is shown in Figure 1.

Following the preprocessing steps in an existing paper [27], we cleaned both GPS dataset and road network dataset, and performed map matching to map the trajectories to the road network. To facilitate trajectory filtering, a spatial quad-tree index is built on trajectory data set.

In the beginning of run-time stage, trajectories can be filtered interactively from spatial and temporal aspects. With a suite of circular graphical filters, user is able to set region of interest by multiple filters. With region of interest settled done, trajectories satisfying certain spatial constraints are filtered. Additionally, a two-layer time filter supports to set constraints in date and time scales. After filtering, the selected trajectories are fed as input to an adaptive DBSCAN clustering algorithm [20], by which O regions and D regions are detected from the origins and destinations of trajectories respectively. OD regions are recognized where O regions and D regions are within a distance threshold.

Traffic flows of O/D, OD regions are extracted and corresponding dynamics are computed. To help user explore OD patterns, our system provides two level visualizations. As an overview, glyphs on the map are designed to show regions' spatial distribution and the average temporal distributions of trajectory number. Meanwhile, statistical distribution, e.g. travel distance and travel time cost are also given. In detail, a refined OD-Wheel visualizes dynamics of traffic volume at finer temporal granularity. Interactions on OD-Wheel support user to compare and correlate temporal dynamic among clusters.

4 FILTER

In this section, we first introduce filter model and then present visual design of spatial and temporal filters.

4.1 Filter Model

Our filter model selects trajectory from spatial and temporal aspects. The filter model is built on a series of atomic queries, which are formalized in Table 1. With these atomic queries, complex filtering can be composed.

From spatial aspect, considering the spatial relationships between trajectory and region, region and region, the atomic spatial queries can be further divided into two classes: location and direction. Six location options are provided according to six possible relationships between trajectory and region: *origin*, *destination*, *origin/destination*, *passing*, *inclusive*, and *exclusive*. Particularly, *exclusive* option provides the function of filtering out trajectories passing a certain area. On the other hand, with two or more filters, direction can be assigned among them to filter trajectories following defined directions.

From temporal aspect, a date-time temporal concept model is used. Not only regular continue time setting, it also allows users to set time constraints in daily periodicity, for example, setting a date range from March 2 to March 5 and time range from 8:00 to 12:00.

The filter model allows users to set constraints iteratively. Complex query can be made base on querying result from previous queries. For example, users can set several location queries to filter trajectories travelling through certain regions.

As mentioned in Section 3.1, the region of interest can be defined by either a spatial filter or multiple filters.

4.2 Filter Design

Similar to TrajectoryLenses [13], we choose circle as the basic filtering shape, whose points on boundary have equal distance to center point. The circular filter is embedded in spatial view so that it is intuitive to directly manipulate the filter in spatial context. There are two design considerations:

Usage Simplicity Parameter tuning is integrated into the circular filter, which is free from keyboard and menu. As Figure 3(a) shows, different functions are waked when hovering in certain regions and corresponding manipulating handles are visible. For example, when hovering on the center of circular filter, the moving function is invoked and a cross mark is visible in the center. Different from parameter setting on a single filter, direction between filters can be assigned by dragging from one to another.

Semantic Visibility For the ease of parameter perception, parameters are explicitly encoded in circular filter. As shown in Figure 4(a), text below informs the radius of underlying region. The right top shows spatial filter with current location constraint (e.g. inclusion constraint here). Figure 4(b) shows filters with different location options. For the assigned direction, an arrow is linked between filters as Figure 4(c) shows.

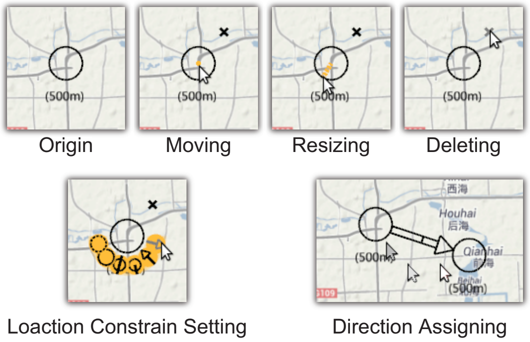

Fig. 3. Interactions on Circular Filter: (a) different function is invoked by hovering on different region. Direction between filters is assigned by dragging from one to another. (b) a part of date and time temporal filter.

A two-level temporal filter is provided: date and time (Figure 5). Users can select any date(s) and time range. Time granularity is 10 minutes. In date part, weekday is colored in white and weekend is green. By dragging and moving, user can define a date range and time range respectively, and those selected are colored in orange. User can cancel current selection by double clicking in blank area.

5 Overview of OD Clusters

In this section, we first introduce the regions extracted from filtered trajectories and then present the visual design of OD overview and detail pattern exploration.

5.1 O/D, OD Region

After filtering trajectories of interested region, we adopt the iterative DBSCAN algorithm [20] on origins and destinations respectively to obtain O clusters and D clusters (**T1**). The convex hull underlying each cluster is detected as O or D region by Graham's scan [9]. Regions within a certain distance (Threshold *D*) are merged into a larger region. Specifically, if the merged region contains both O and D regions, it is an OD region.

5.2 Overview

Overview of regions consists of spatial overview (**T2**) and statistical overview (**T3**). As Figure 6 shows, two types of spatial overview are provided. Figure 6(a) shows the original clusters' convex hulls, which gives precise spatial description of regions. However, constrained by absolute spatial shape, it is not convenient to encode more information upon it. Hence, as Figure 6(b) shows, spatial view with glyphs is designed as a trade-off between abstract and spatial space.

As Figure 7 illustrates, each region is represented by a square glyph. The region's traffic volume is mapped to the size of square. The square's frame color is used to distinguish whether the region is O or D or OD region. Taking O regions as examples, a sequential blue list are assigned to them from dark to light in descending traffic volume order. So is the red list for D regions and green list for OD regions. To make it more intuitive, an arrow in the right-bottom explicitly encodes the direction of traffic flow, such as an going-out direction in O region glyph. For each region, to emphasize on the traffic flow related to other detected regions, the volumes of related traffic flows are visualized in color dithering manner inside the glyph. That is the ratio of certain color inside one glyph is proportional to the traffic volume related to corresponding region. Additionally, the average traffic volume distribution in a day can be unfolded by clicking the dot handle in the right-top. From left to right, the average traffic volume distribution is displayed in hour granularity from 00:00 to 24:00, smaller value with larger lightness. Every four hour a gray vertical line is drawn to help with hour reading. When folded, the number of trajectories is labelled aside to the dot handle. Particularly for OD glyph, there are two dots, one for the coming-in and other for the going-out flow.

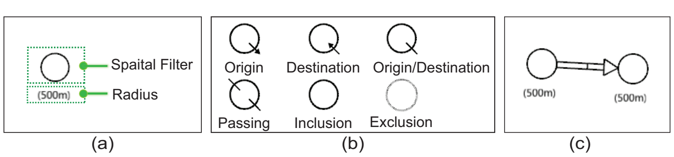

Fig. 4. Circular Filter Design: (a) circular filter glyph: radius and spatial filter with location constraints are explicitly encoded. (b) six types of spatial filters (the left-top circle in glyph) with different location constraints. (c) arrow encoding the direction between circular filters.

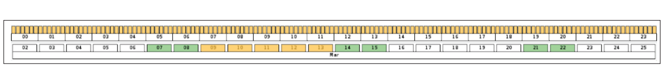

Fig. 5. Two-level Temporal Filter

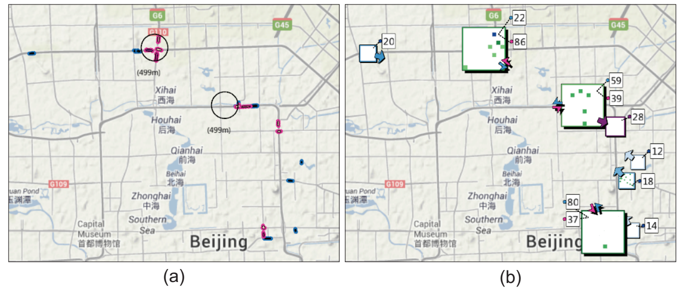

Fig. 6. Spatial Overview: (a) regions are visualized in convex hull. (b) glyphs are used to represent the detected regions.

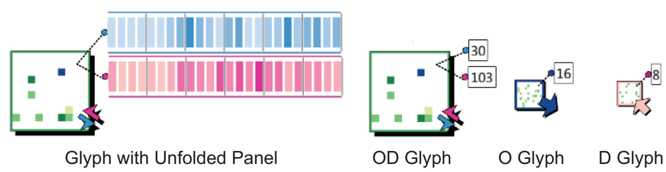

Fig. 7. Glyph Design for O/D, OD Regions

As Figure 8 shows, the statistical overview contains the traffic volume distribution over time, which can be splitted up into weekdays and weekends by mouse right clicking. Another two bar plots (Figure 8(c)) show the distribution of average travel distance and travel time. Aware of extreme values, we wrap the extreme values and count them as a whole in the two ends of box plot.

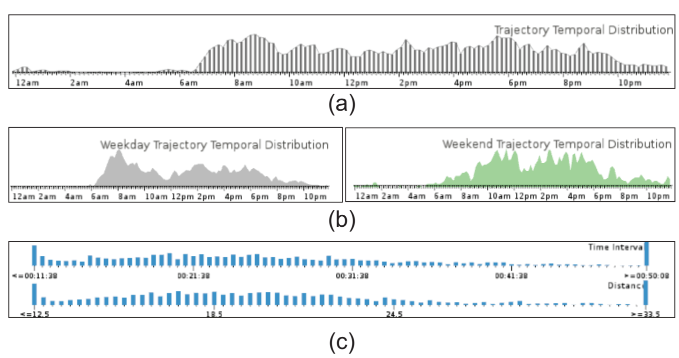

Fig. 8. Statistical Overview: (a) traffic volume distribution over view; (b) traffic volume distribution on weekdays and at weekends; (c) travel distance and travel time histogram

6 ADAPTED OD-WHEEL

Extended from exploration OD clusters related to a central region, OD-Wheel [17] is adapted to explore the OD clusters detected from interested trajectories, which is not restricted to OD clusters related to a central region. Given the O/D, OD regions, the adapted OD-Wheel is designed to explore dynamic change of clusters (**T4**) and compare OD patterns among clusters (*T5*).

6.1 Visual Design

The main idea of OD-Wheel is to warp a part of linear view to circular one. O, D clusters are placed in descending traffic volume order from two ends of circle respectively. Particularly in this work, OD region is decomposed into a pair of O and D clusters, which counts the in and out traffic flow to the region respectively. A link is built between the pair, whose color is consistent with the OD glyph in spatial overview. Additionally, an arrow towards/outwards the circular center is used to distinguish O and D clusters further. And its size is proportional to the cluster's traffic volume. In much more detail, the dynamic traffic volume of each cluster is visualized as bar plot along radial time axis. Sharing the same axis with circular view, undistorted linear layout benefits more precise dynamic analysis. For example, as Figure 9 shows, analysis on travel time cost or dynamic traffic volume can be performed in linear plot. However, it is easy to extend to analysis on other dynamic variations.

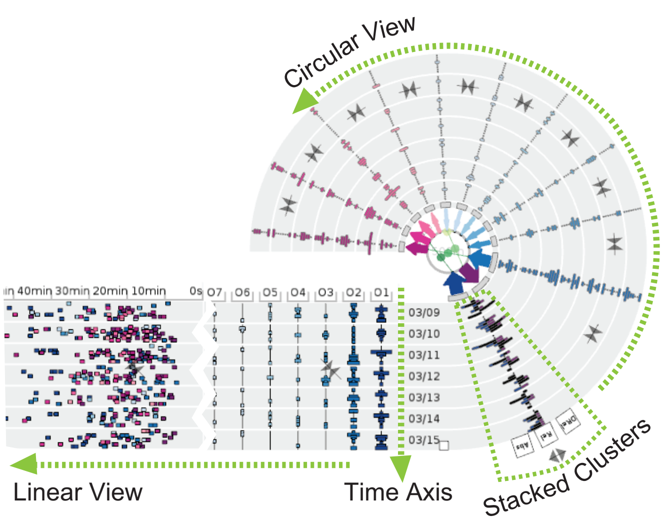

Fig. 9. Visual Design of OD-Wheel: linear and circular view displays the temporal distribution of travel flow volumes. Two possible dynamic linear visualizations are in the linear view.

6.2 Interactions on OD-Wheel

Several interactions are developed for easy comparison among clusters. A cluster can be relocated by dragging-and-moving the gray annulus around the circle. Clicking black button (><) between pair of neighbouring clusters stacks the two clusters to a common middle line, as the right bottom part in Figure 9 shows. To facilitate the comparison, differences between the two clusters along time are explicitly encoded by black bars. By clicking the black button (<>) outside the circle, the stacked bars can be unstacked.

When mouse hovering on bars, its time interval pops out as hint. Circular view and linear view cooperate in brush-and-link manner. Hovering on the arrow glyph in circular view highlights corresponding clusters in the linear view. Similarly, lasso selection on trajectories in linear view highlights corresponding ones in circular view and spatial view.

7 IMPLEMENTATION

With the quad-tree index built on trajectory data, spatial filter on trajectories is decomposed into three steps: coarse filter by cells of quad-tree, trajectory candidates loading in memory and fine filter by the circular filter setting. To ensure the smoothness of interactions, the first N trajectories (e.g. $N = 100$) are queried and rendered when dynamic interactions. For example, during the process of resizing filter, the query mechanism filters the first N trajectories. Only when settling down, the whole dataset is queried and the complete query result is returned.

Our system is implemented using C++/QT, using the OpenGL and QGraphicsView rendering framework. The experiments are conducted on an Intel(R) Core(TM)2 2.66GHz Laptop with 4GB RAM and a NVIDIA Geforce GTX 470 GPU. The preprocessing is run off line. After the preprocessing, system supports interactive visual displays and user interactions.

8 CASE STUDY

We apply the system to real taxi trajectories to show its effectiveness.

Our GPS dataset is a real taxi dataset recorded in the city of Beijing in 24 days, from March 2nd to 25th, 2009. Estimated from a government report [1], they include 43% of all licensed taxis in Beijing, and account for 7% of the traffic flow volume within Beijing's 4th Ring. We only use trajectories carrying passengers. After preprocessing, the final dataset is 12.8 GB.

8.1 Statistical Overview of Interested Regions

The system provides a statistical overview of OD traffic volume distribution **T3**.

Using interactive filter, user is able to select trajectories related to interested regions. As Figure 10(a) shown, four interested regions are selected as observing center respectively. For each region, trajectories are filtered further by other two filtering options: coming in or going out of the region, on weekdays or at weekends, from March 9^{th} to March 15^{th} 2009 which covers a whole week.

Tow line plots show the distribution of normalized average trajectory number per 10 minutes on a weekday and weekend respectively. In Figure 10(b), different temporal distribution patterns can be observed: there are lots of taxis coming into *A* from 8:00 to 9:00 on weekdays, while this morning peak disappears in another three plots of *A*. On the contrary, *B* gets its morning peak in the going-out direction and evening peak in coming-in direction on weekdays. For both *A* and *B*, their temporal distributions at weekends are more casual. Different from *A* and *B*, either *C* or *D* has similar temporal trends on weekdays and at weekends. Specially, for *C*, there is a quite early morning peak in coming-in direction, nearly 6:00, which can be explained by experience that people may take taxi to airport in *C* to catch early flight. Taxis going out from *C* mainly become active from 10:00. Similar to *C*, *D* is where Beijing-West Railway Station locates. The temporal distribution in *D* spans a little longer than its counterpart in *C*, coming-in taxi flow is from around 5:00 to 22:00 and going-out taxi flow is from 6:00 to 24:00.

8.2 OD Spatial Overview of Interested Region

The system provides a spatial overview of interested region **T2**, such as a crossing region or a road. In this case, we demonstrate two case studies, one for an interested crossing and the other for an interested road.

We set the interested region at a crossing in the Beijing, which is drawn as black ring in Figure 11(a). Based on the trajectories passing through this region, several O/D, OD regions are detected as Figure 11(a) shows. The two biggest OD regions are around the Beijing West Railway Station and Beijing South Railway Station. That is, taxis passing through this crossing are likely to origin or reach in this two regions. Besides these two OD regions, there are three O regions extracted. One of them is around the Beijing Railway Station (the right glyph of Figure 11(a)). Figure 11(b) shows a close up view of the glyph. Dense green color inside the glyph indicates that many of the

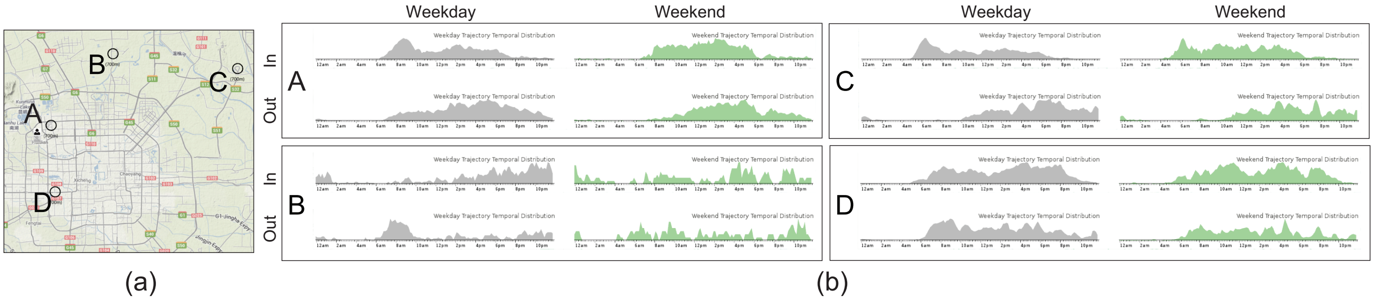

Fig. 10. Trajectories Temporal Distribution: (a) four central regions are set to filter trajectories of interest respectively, each with two options (O or D) and two temporal options (weekday or weekend), from March 9^{th} to March 15^{th}. (b) line plot visualizes the temporal distribution of trajectories from each filtering setting.

trajectories from Beijing Railway Station passing the interested region reach the Beijing West Railway Station. The other O region locates at one of the biggest commercial centers (the light blue one with 18 related trajectories). As Figure 11(c) shows, some of trajectories end at the railway station (the dark green) and some end at others (the light green).

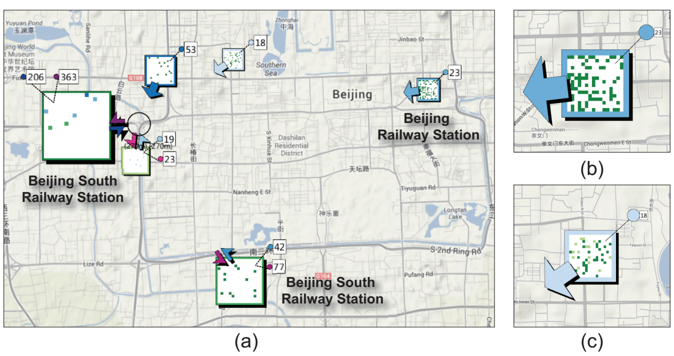

Fig. 11. OD Spatial Distribution of a Crossing Region: (a) the detected O/D, OD regions. (b) close-up view of O region at Beijing Railway station. (c) close-up view of O region at one commercial center.

Besides defined by a single spatial filter, the interested region can also be composed of multiple filters. Figure 12 shows the OD spatial overview of an interested road, which is defined by two filters on the 2^{nd} ring road, Beijing. The O/D, OD regions of trajectories travelling through this road are detected and visualized in Figure 12(a) and Figure 12(b). Most of the regions are along the 2^{nd} ring. Another two OD regions locate at the two terminals of Beijing Airport (Figure 12(b)). Unfolding the average distributions of regions marked by 1, 2 as Figure 12(c) shows, we found that there is an early morning peak (6:00-9:00) and evening peak (18:00 -21:00) at 1 and 2 respectively. Unfolding coming-in and going-out traffic flows at airport, the two terminals have similar OD pattern over day. For the coming-in traffic flow, it becomes active in early morning and keeps active during the day time. On the contrary, the out-going flow becomes active from the noon until evening.

8.3 Detail Exploration on Volume Dynamic Change

Besides of overview of the OD pattern, the system supports to explore the OD volume dynamic change in deail by the adapted OD-Wheel (**T4**).

In this case, the interested region is set at the west-north corner on the 4^{th} ring road. 5124 Trajectories passing this region are filtered from Mar. 9 to Mar. 11. The spatial distribution of O/D, OD regions are as Figure 13(a) shows. The dark green OD region has the largest traffic volume. For all the other regions, there are dense dark green inside the cluster glyphs. That is, many trajectories of regions passing

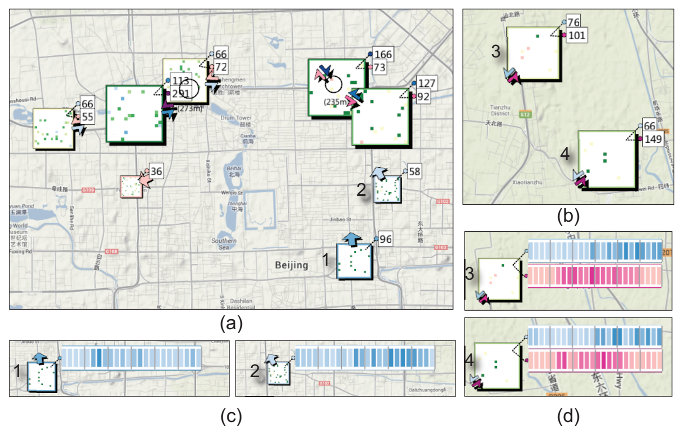

Fig. 12. OD Spatial Distributio of a Road: (a) the detected O/D, OD regions in down-town (b) the two detected OD regions at Beijing Airport. (c) unfolded distribution panel of 1 and 2 regions in (a). (d) the unfolded distribution panel of OD regions marked 3, 4.

the interested regions start or end at the largest OD region. Especially, the second largest OD region is in the east-north to the largest one with a short distance. Unfolding the in and out traffic volume distribution of the largest OD region from 00:00 to 24:00, the average distribution over day can be observed respectively. Examining in OD-Wheel for detail, the pair of in and out traffic volume has definitely different distribution shapes in the linear view. To compare them clearly, the two are stacked. The black difference bars clearly show that there is larger out rather than in traffic flow at beginning and then larger in traffic flow in the afternoon and evening.

8.4 OD Pattern Dynamic Exploration

With linear view integrated in OD-Wheel, the system supports to explore the OD dynamic pattern **T5**.

In this work, we take the travel time cost as an example of OD dynamic pattern, while our method can be easily extended to other measurements, such as travel distance. In this case, we set the interested region at the north-west corner on the 2^{nd} ring road, which is one of the biggest transportation hub in Beijing. 5762 trajectories are filtered from Mar. 9 to Mar. 10. As Figure 14(a) shows, the largest OD region is around the transportation hub, which conveys over 1000 taxis in and out every day. Others regions are significantly smaller than the largest one. Comparing the traffic volume of the largest OD region in the OD-Wheel, we found that the in-direction traffic flow is larger than the out-direction in the morning and inversely in the afternoon. The travel time cost of the two directions are visualizes as dot plots in the linear view (Figure 14(b)). Averagely, most of the travel time cost from or to this region is within 30 min. However, in longer travel time range, the in-direction spans wider than the one of out-direction. The

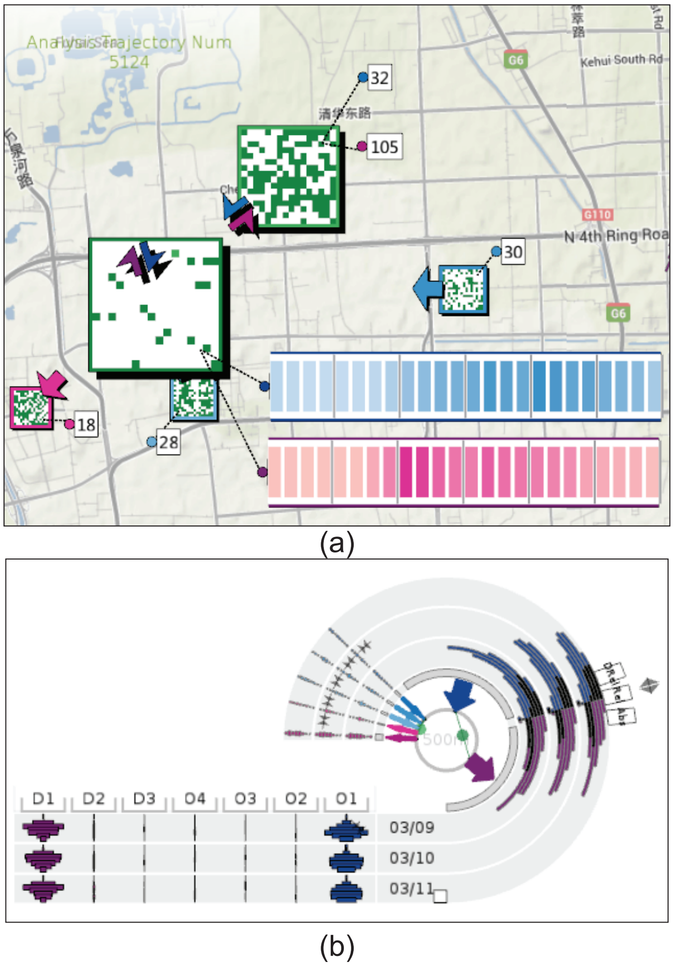

Fig. 13. Explore Traffic Volume with OD-Wheel: (a) O/D, OD regions detected at the interested region, with unfolded in and out traffic flow distribution of the largest OD region. (b) in and out traffic volume comparison in the OD-Wheel.

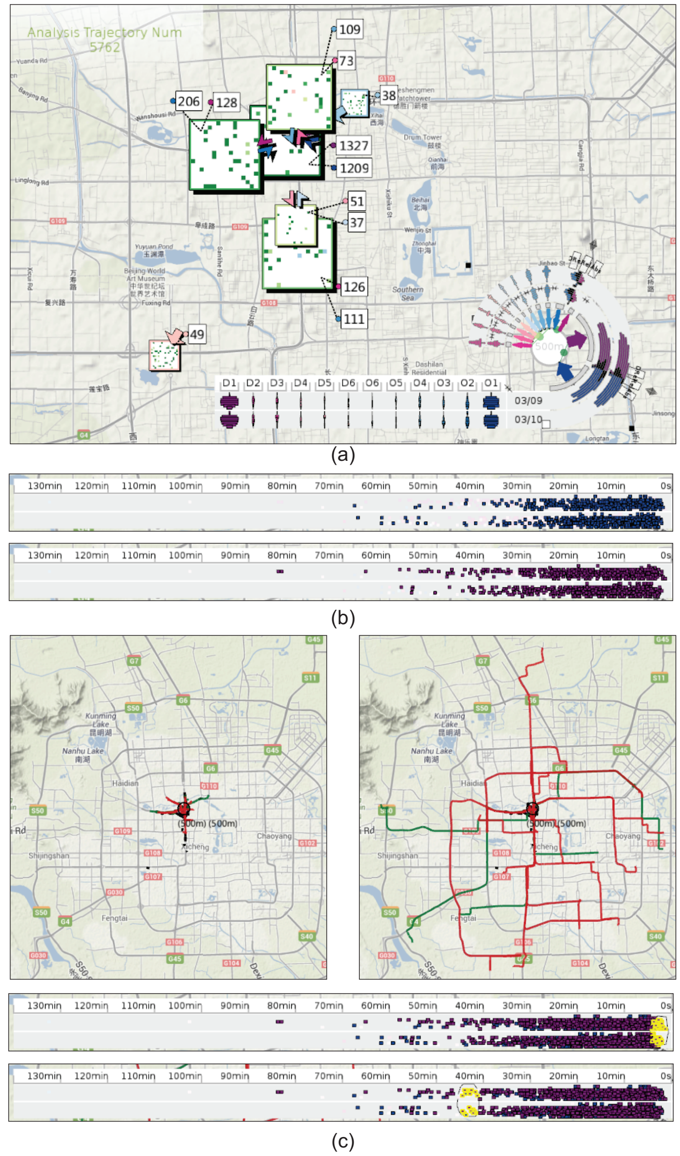

Fig. 14. Exploration on Travel Time Cost: (a) OD regions extracted from trajectories of a transportation hub in the north-west corner on the 2^{nd} ring. (b) travel time cost distribution of the in/out traffic flows to/from the larger OD region. (c) trajectories with short and large time cost.

link-and-brush function between linear view and map view supports users to check trajectories in detail. As Figure 14(c) shows, trajectories with small time cost and with large time cost are selected in the linear viw and plotted on the map. Those with small time cost (within 10 min) travel in very short distance and those with large time cost (more than 40 min) reach the 5^{th} ring of Beijing.

9 Discussion

In this work, OD-Wheel is applied to analyse dynamic OD patterns extracted from trajectories. In essence, it can be potentially generalized as a visualization that supports to compare time series in circular view and explore related dynamic attributes in the linear view. For example, considering the traffic data at each RFID cell [28] as time series data, OD-Wheel can be used to help with the comparison among different cells. Additionally, for OD-Wheel design, it is flexible to customize the linear view according to specific analysis tasks provided sharing the same time axis with circular view.

However, some drawbacks are imported when using the circular layout. Firstly, the number of time series in OD-Wheel is suggested within 10, limited by the visual space for each time series as well as the color. Secondly, circular layout arises the intrinsic distortion. Although we encodes the value to arc length instead of angle, the distortion changes as the radius changes.

In the map view, we design a glyph on the map as the balance between spatial space and abstract space. However, we take the centroid of extracted convex hull as the center position of the glyph. In some cases, heavy visual occlusion arises when two or more glyphs come close with each other. Some re-locate layout algorithm can be applied to alleviate this problem, which we have not considered in current work.

10 Conclusion

In this paper, we present a visual analytic system which supports to explore the OD patterns of interested regions. Instead of global OD pattern analysis, we tackle this problem starting from trajectory filtering. With a circular filter, the system supports to define spatial constraints as interested region and filter related trajectories. Based on these trajectories, OD clusters are extracted and corresponding regions are detected. To analyse the traffic pattern of those OD regions, we adapt a visual design, OD-Wheel, to support the comparison among regions, which are not limited to a central region as before. With effective interactions and intuitive visual hints in comparison, the adapted OD-Wheel is capable of comparing dynamic patterns in detail. Applying the method to real Beijing taxi GPS trajectories, the system's effectiveness is demonstrated by several cases, from the statistical and spatial overview on OD regions to detail analysis of OD patterns.

There are some possible research directions of this work. One is that OD-Wheel can be generalized as a visual tool to explore time series data. More interactive functions could be added on OD-Wheel. For example, when the number of compared time series increases, those time series with small values would occupy a small visual space, which is not easy to percept the value. Hence, a magnifying function can be added. On the other hand, the traffic interchange flow among multiple regions will be considered in the future work. OD flow or matrix can be integrated into OD-Wheel so that it is possible to explore the traffic flow between pair of two certain regions.

References

[1] Beijing transportation research center: Annual report of beijing transportation development, 2010.

[2] G. Andrienko and N. Andrienko. Spatio-temporal aggregation for visual analysis of movements. In *IEEE Symposium on Visual Analytics Science and Technology*, pages 51–58, 2008.

[3] N. Andrienko and G. Andrienko. Spatial generalization and aggregation of massive movement data. *IEEE Trans. Vis. Comput. Graph.*, 17(2):205 –219, 2011.

[4] M.-A. Aufaure-Portier and C. Bonhomme. A high level visual language for spatial data management. pages 325–332, 1999.

[5] I. Boyandin, E. Bertini, P. Bak, and D. Lalanne. Flowstrates: An approach for visual exploration of temporal origin-destination data. *Comput. Graph. Forum*, 30(3):971–980, 2011.

[6] N. Cao, Y.-R. Lin, X. Sun, D. Lazer, S. Liu, and H. Qu. Whisper: Tracing the spatiotemporal process of information diffusion in real time. *IEEE Trans. Vis. Comput. Graph.*, 18(12):2649–2658, 2012.

[7] N. Ferreira, J. Poco, H. Vo, J. Freire, and C. Silva. Visual exploration of big spatio-temporal urban data: A study of new york city cab trips. *IEEE Trans. Vis. Comput. Graph.*, 19(12):2149–2158, 2013.

[8] K. Fishkin and M. C. Stone. Enhanced dynamic queries via movable filters. pages 415–420, 1995.

[9] R. Graham. An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters*, 1972.

[10] D. Guo. Flow mapping and multivariate visualization of large spatial interaction data. *IEEE Trans. Vis. Comput. Graph.*, 15(6):1041 –1048, 2009.

[11] H. Hochheiser and B. Shneiderman. Dynamic query tools for time series data sets: timebox widgets for interactive exploration. *Information Visualization*, 3(1):1–18, Mar. 2004.

[12] D. Holten and J. van Wijk. Force-directed edge bundling for graph visualization. *Comput. Graph. Forum*, 28(3):983–990, 2009.

[13] R. Krüger, D. Thom, M. Wörner, H. Bosch, and T. Ertl. Trajectorylenses - a set-based filtering and exploration technique for long-term trajectory data. *Comput. Graph. Forum*, 32(3):451–460, 2013.

[14] H. Liu, Y. Gao, L. Lu, S. Liu, H. Qu, and L. Ni. Visual analysis of route diversity. pages 171–180, 2011.

[15] W. Liu, Y. Zheng, S. Chawla, J. Yuan, and X. Xing. Discovering spatio-temporal causal interactions in traffic data streams. In *Proc. ACM SIGKDD*, pages 1010–1018, 2011.

[16] X. Liu, Y. Hu, S. North, T.-Y. Leea, and H.-W. Shen. Correlatedmultiples: Spatially coherent small multiples with constrained multidimensional scaling. Technical report, Ohio State University, 2013.

[17] M. Lu, Z. Wang, J. Liang, and X. Yuan. Od-wheel: Visual design to explore od patterns of a central region. In *Proceedings of IEEE Pacific Visualization Symposium (PacificVis 2015 Notes)*, pages 14–17, 2015.

[18] M. Lu, Z. Wang, and X. Yuan. Trajrank: Exploring travel behaviour on a route by trajectory ranking. In *Proceedings of IEEE Pacific Visualization Symposium*, 2015.

[19] P. McLachlan, T. Munzner, E. Koutsofios, and S. North. Liverac: Interactive visual exploration of system management time-series data. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1483–1492. ACM, 2008.

[20] G. Pan, G. Qi, Z. Wu, D. Zhang, and S. Li. Land-use classification using taxi gps traces. *IEEE Transactions on Intelligent Transportation Systems*, 14(1):113–123, 2013.

[21] C. Peng, X. Jin, K.-C. Wong, M. Shi, and P. Li. Collective human mobility pattern from taxi trips in urban area. *PLoS ONE*, 7(4):e34487, 2012.

[22] D. Phan, L. Xiao, R. Yeh, and P. Hanrahan. Flow map layout. In *IEEE Symposium on Information Visualization*, pages 219 – 224, 2005.

[23] A. Rae. 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(3):161 – 178, 2009.

[24] G. Robertson, R. Fernandez, D. Fisher, B. Lee, and J. Stasko. Effectiveness of animation in trend visualization. *IEEE Trans. Vis. Comput. Graph.*, 14(6):1325–1332, 2008.

[25] W. Thompson and S. Lavin. Automatic generation of animated migration maps. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 33:17–28, 1996.

[26] J. van Wijk and E. van Selow. Cluster and calendar based visualization of time series data. In *IEEE Symposium on Information Visualization*, pages 4 –9, 1999.

[27] Z. Wang, M. Lu, X. Yuan, J. Zhang, and H. van de Wetering. Visual traffic jam analysis based on trajectory data. *IEEE Trans. Vis. Comput. Graph.*, 19(12):2159–2168, 2013.

[28] Z. Wang, T. Ye, M. Lu, X. Yuan, H. Qu, J. Yuan, and Q. Wu. Visual exploration of sparse traffic trajectory data. *IEEE Trans. Vis. Comput. Graph.*, 20(12):1813–1822, 2014.

[29] M. Weber, M. Alexa, and W. Muller. Visualizing time-series on spirals. In *IEEE Symposium on Information Visualization*, pages 7 –13, 2001.

[30] J. Wood, J. Dykes, and A. Slingsby. Visualization of origins, destinations and flows with OD maps. *Cartographic Journal*, 47:117–129, 2010.

[31] J. Zhao, F. Chevalier, and R. Balakrishnan. Kronominer: using multi-foci navigation for the visual exploration of time-series data. In *Proc. of ACM SIGCHI 2011*, pages 1737–1746, 2011.