Eurographics Conference on Visualization (EuroVis) 2013
B. Preim, P. Rheingans, and H. Theisel
(Guest Editors)

Volume 32 (2013), Number 3

Visualizing Interchange Patterns in Massive Movement Data

Wei Zeng[1,2], Chi-Wing Fu[2], Stefan Müller Arisona[1,2], Huamin Qu[3]

[1]Future Cities Laboratory, ETH Zurich, Switzerland
[2]Nanyang Technological University, Singapore
[3] The Hong Kong University of Science and Technology, Hong Kong

Abstract

Massive amount of movement data, such as daily trips made by millions of passengers in a city, are widely available nowadays. They are a highly valuable means not only for unveiling human mobility patterns, but also for assisting transportation planning, in particular for metropolises around the world. In this paper, we focus on a novel aspect of visualizing and analyzing massive movement data, i.e., the interchange pattern, aiming at revealing passenger redistribution in a traffic network. We first formulate a new model of circos figure, namely the interchange circos diagram, to present interchange patterns at a junction node in a bundled fashion, and optimize the color assignments to respect the connections within and between junction nodes. Based on this, we develop a family of visual analysis techniques to help users interactively study interchange patterns in a spatiotemporal manner: 1) multi-spatial scales: from network junctions such as train stations to people flow across and between larger spatial areas; and 2) temporal changes of patterns from different times of the day. Our techniques have been applied to real movement data consisting of hundred thousands of trips, and we present also two case studies on how transportation experts worked with our interface.

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [Information Interfaces and Presentation]: User Interfaces—Evaluation/Methodology I.3.8 [Computer Graphics]: Application—Geographical Visualization

1. Introduction

Movement data is generally a collection of object trajectories over time and space. In its simplest form, an object is denoted as a single point with trajectory represented as an ordered sequence of observations [HE02]. From this, we can estimate information on individual objects, e.g., speed and direction, and also aggregated information on the entire data, e.g., density distribution and movement patterns [AAPS08].

A number of advanced data acquisition technologies have been developed recently for capturing movement data: location-positioning by cell phones and GPS, personalized user-tagged cards for public transportation, and video analysis for people and vehicle flows. These technologies benefit many scientific research disciplines, for example, the study of animal habitats and their spatial distribution [Gan01], the reconstruction of traffic flows from traffic sensors [SvLM11], and functional road models for traffic simulation [WSL12]. However, such advancement also increases the data set size, thus making the problem of visualizing and exploring movement data to be nontrivial. Traditional methods [Kwa00, KW04], which directly plot the object trajectories in 2D/3D, could simply fail because of visual cluttering and occlusion.

To address these issues, there are two major visualization approaches [AAD*08]: 1) *pattern extraction*, which applies knowledge discovery methods [LKI05, GNPP07] to find out motion patterns; and 2) *data aggregation*, which groups locations into regions and summarizes the movement data in a regional basis [Guo09, AA11]. This work considers both strategies. In particular, we are interested in studying and visualizing a high-level aggregated motion pattern:

> *Interchange pattern*, which describes how moving objects redistribute when entering and passing through a junction node in a traffic network.

Our formulation also considers the study of interchange patterns at different scales: train stations in a metro system, crossroads in a road network, or regional zones in a city.

Interchange patterns are a highly valuable means not only for unveiling mobility patterns, but also for assisting transportation planning. For instance, interchange information can help reveal the road junction utilization and suggest crossroad redesign, e.g., adding a fork. A similar situation is also shared by the case of train stations, where interchange patterns can help improve the interior design of routes and platform connections within a station. At city scale, interchange patterns of people flow can help indicate longer distance trips or detours that are undertaken by some people, thus suggesting the transportation efficiency for enhancing the road network design.

To support efficient visualization of interchange patterns that emerged from massive movement data, we propose a novel visual representation, namely the *interchange circos diagram*, for presenting the redistribution of people at junction nodes. This visual design is adapted from the circos figure [KSB*09], which was invented for examining the mutual relationships between genomes. Incorporated with various advices from domain experts, we revise and customize the circos figure for presenting passenger interchange: a fly-over ring to denote the junction node itself, bi-directional bundling to reduce visual cluttering, and an optimized color assignment on linkages to enhance the visual connections between neighboring interchange circos diagrams. Our visualization techniques have been applied to real world movement data consisting of hundred thousands of trajectories, and two case studies on how transportation experts applied our method are also presented.

The major contributions of this work are:

- First, we propose to visualize interchange patterns that emerged from massive movement data. To the best of our knowledge, this is the first work designed for visualizing and exploring spatio-temporal interchange patterns emerged from large volume of trajectory data.
- Second, we design a novel visual representation, the *interchange circos diagram*, to present the redistribution of moving objects in a compact manner. Various design considerations, e.g., visual cluttering, visual connections, and statistics summarization, are considered.
- Lastly, we develop an interactive interface that can process massive movement data (with pre-computed information) in real-time and support interchange visualization with various user interactions. Moreover, we experiment it with real world movement data and two case studies.

2. Related Work

This section reviews the following areas of related research:

Geovisualization. Geovisualization aims at developing interactive visual tools to explore and analyze spatial data that can be mapped geographically [MK01]. As it is a broad and extensively studied field, we discuss only a few works here. Kwan [Kwa00] presented a family of GIS-based methods that can simultaneously address both the spatial and temporal dimensions of human activity-travel patterns. Kapler and Wright [KW04] developed GeoTime to show movement data as 2D paths in a 3D space-time domain. Wood et al. [WDSC07] combined interaction and visual encoding to visualize large amount of multifarious spatiotemporal data. Zeng et al. [ZZA*12] designed an interactive system to explore spatiotemporal trajectory data with mobile devices.

Motion Patterns. Motion pattern analysis aims at discovering combined behavior of a group of moving objects in space and time. Laube et al. [LKI05] defined four spatio-temporal motion patterns, i.e., flock, leadership, convergence, and encounter, and discussed approaches to mining them from movement data. Gudmundsson et al. [GvKS07] later devised approximation algorithms based on computational geometry methods to speed up the mining. Giannotti et al. [GNPP07] defined the notion of trajectory pattern mining and proposed regions-of-interests-based methods to extract frequent patterns from trajectory data. Dodge et al. [DWL08] suggested some potentially useful dimensions towards a taxonomy of describing and classifying motion patterns: generic vs behavioral, primitive vs compound, and group vs individual. Pelekis et al. [PKM*07] formulated distance operators to compute trajectory similarity and proposed different types of similarity queries based on various motion parameters. Rinzivillo et al. [RPN*08] developed a progressive clustering method to aid the visual exploration of groups of similar (and dissimilar) trajectories. Liu et al. [LGL*11] studied route diversity in real trajectory data, and proposed visual encoding schemes to display, compare, and evaluate routes.

Movement Data Visualization. Traditional methods [Kwa00, KW04] generally plot trajectory paths directly in 2D/3D according to the geographical context. This is sufficient for small amount of trajectories, but when the data set becomes large and complex, visual cluttering and occlusion problems could appear.

Based on kernel density estimation, density maps were proposed as a mean to summarize large amount of trajectory paths, so that we can overview the distribution of moving objects without visual cluttering and occlusion problems. Willems et al. [WvdWvW09] proposed a density-map-based interface to visualize vessel movements: large kernels to overview spatial utilization and reveal vessel highways, and small kernels to show speed variations of individual vessel. Scheepens et al. [SWvdWvW11, SWvdW*11] further improved the method by filtering the trajectories and enabling customized versatile exploration of the data using multiple density fields. Rather than using kernel density, Hurter et al. [HTC09] developed a multidimensional visualization tool based on a brush/pick/drop paradigm for users to explore large amount of aircraft trajectories across multiple views, whereas Guo et al. [GWY*11] proposed a visual analytic interface to explore traffic data in a microscopic scale.

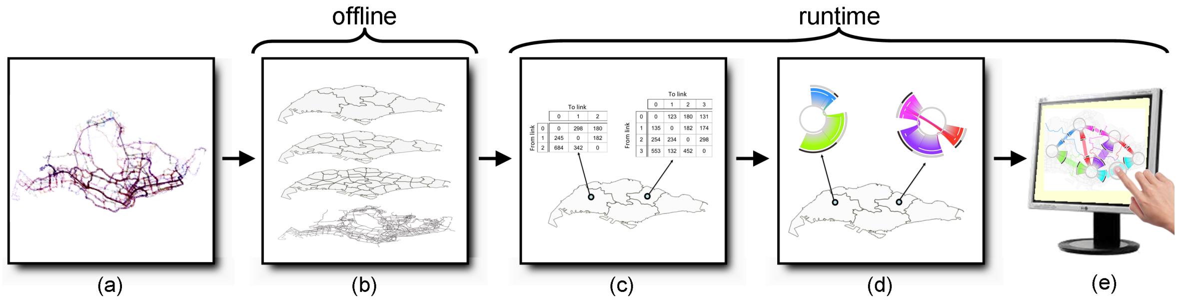


Figure 1: *System workflow: from (a) a set of raw trajectory paths, to (b) traffic networks of different spatial scales, (c) interchange statistics, (d) interchange circos diagrams per junction node, and (e) our interchange visualization with user interaction.*

Other than kernel density methods, Andrienko and Andrienko [AA08] proposed various approaches to aggregate movement data over space and time, and developed a variety of interaction methods to aid the visual exploration in combination with aggregation. Guo [Guo09] adapted a graph partition method to construct a hierarchy of geographical regions and applied clustering and visualization methods to analyze county-to-county migration data of people. Andrienko and Andrienko [AA11] proposed a novel way of partitioning spatial regions by extracting characteristic points from the movement data and then grouping them according to spatial proximity. Tominski et al. [TSAA12] proposed to visualize trajectory attribute data; they extracted the functional dependency of the attributes, and then stacked up 3D trajectory bands that are color coded by the corresponding attributes.

This paper focuses on a novel aspect of visualizing and analyzing massive movement data, i.e., *interchange patterns*, which has not been explored in any previous work as we are aware of. In particular, we designed a new visual representation, the *interchange circos diagram*, for visualizing interchange patterns, and developed also a working interface for exploring these patterns across both space and time.

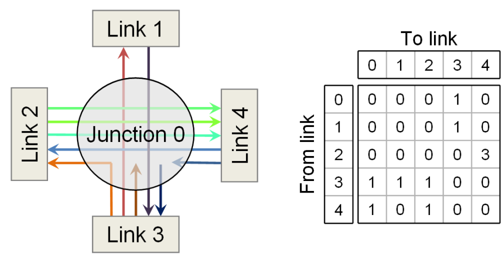
3. Overview

This section first presents a formal definition on interchange patterns, and then overviews our system workflow.

3.1. Formal Definition: Interchange

An interchange pattern at a junction basically describes how moving objects redistribute when they go through the junction. Given a traffic network modeled as an undirected graph, say $G = (V,E)$, where V is the set of (junction) nodes in G and E the set of edges connecting neighboring nodes in V. When a moving object passes through a junction node, say $v \in V$, whose valency is n, it has $n+1$ possible ways of entering the node. This is because it may come from v's n connecting links, or from the dominion of junction v itself; these are the possible sources. Likewise, there are also $n+1$ possible ways (sinks) of leaving junction v.

Hence, given the trajectory data, we first can identify a subset of trajectories that go through each node in V. Then, we can determine the incoming and outgoing links of each trajectory across a node, and summarize the interchange information at the node as a $(n+1)$-by-$(n+1)$ matrix, which counts all the possible routes of going through the node.



From link \ To link	0	1	2	3	4
0	0	0	0	1	0
1	0	0	0	1	0
2	0	0	0	0	3
3	1	1	1	0	0
4	1	0	1	0	0

Figure 2: *The interchange information (ten trajectories) at this junction can be summarized as a* 5*-by-*5 *matrix.*

Figure 2 shows an example of a junction node with four links and ten trajectories. We can summarize its interchange statistics as a 5-by-5 interchange matrix. Note that the diagonal elements in the matrix are all zeros because we assume that no trajectories revert back to the same link.

3.2. System Workflow

Our system workflow consists of the following computational steps, see also Figure 1:

- Starting from the raw trajectory paths (Figure 1(a)), we first build a traffic network in the form of a undirected graph. It can be a road-level network (Figure 1(b)(bottom)), a city-scale network (Figure 1(b)(top)), or a series of region-scale networks in-between. As for the finest-scale network, we can reconstruct it by examining the raw trajectory paths, while for the coarser networks, we can either reconstruct them by hierarchical clustering, such as those in [Guo09, AA11], or obtain the network structure directly from the domain experts.
- Then, for each traffic network, we determine per link (between pairs of neighboring junction nodes) two sets of trajectories (per movement direction along the link) that go

through the link. Next, we partition the total time period covered by the the trajectories into equal time intervals, say 15 minutes, and precompute an interchange matrix for each time interval per junction node.

- After the user interactively chooses a period of time over a day, our system can retrieve and sum up the interchange matrices corresponding to the related time intervals that made up that time period. By this, we can quickly produce summarized interchange matrix (Figure 1(c)) at any junction upon user request. After that, an interchange circos diagram is constructed from the matrix and presented in the visualization (Figure 1(d)), see Section 4.
- Lastly, our interface supports also a family of visualization and user interaction techniques to explore various aspects of the interchange patterns, see Section 5.

4. Our Interchange Circos Diagram

Transportation domain experts expect the following information when examining interchange patterns: (1) absolute and relative flow volumes across different pairs of links at a junction, (2) ratio of total incoming and outgoing flow volumes of each link, (3) flows starting/ending at the junction itself, (4) flow directions, (5) correspondence to the geographical nature of the data, and (6) temporal and spatial variations of the interchange patterns. Hereby, we design a novel visual representation to capture these features.

This section first presents the idea of the original circos figure, and then develops it into the interchange circos diagram to present interchange patterns. Then, we present how the interchange circos diagram is implemented, and compares it against existing visual representation.

4.1. Circos Figures

The circos figure was invented by Krzywinski et al. [KSB*09] for examining the mutual relationship among genomes. After constructing a two-dimensional table of relationships, such as similarity and difference, among pairs of elements in the genomes, its basic idea is to present the pair-wise data matrix in a circular ideogram layout with ribbons that connects related elements, see Figure 3 for examples. Other than genome visualization, the circos figure was also adopted by Bostock et al. [BOH11]

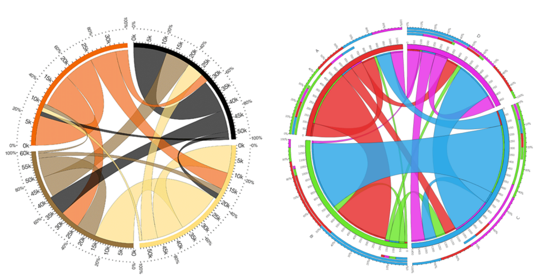


Figure 3: *Example circos figures.*

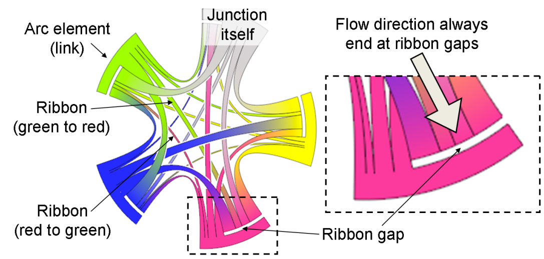


Figure 4: *An initial design of interchange circos diagram.*

for web visualization, and another related visual metaphor that shares similar characteristics is the contingency wheel [AGMS11].

4.2. Initial Design: Interchange Circos Diagram

To develop our interchange circos diagram from circos figures, the very first step is to map the interchange information to the various visual components in a circos figure. First, we map the connecting links at a junction node (including the junction itself) as *arc elements* around the figure's boundary, and vary the angular size of these arc elements according to the total flow volume across the links, see Figure 4. Then, we join the arc elements with *curved ribbons* and vary the ribbon width to present the flow volume.

Moreover, we sort and render the ribbons from back to front to emphasize the flows with larger volume, and employ haloes [ARS79, IG98] to visually emphasize the occlusions between intersecting ribbons. Next, we assign a unique color to each arc element (see Section 4.4), and specifically assign grey to indicate the junction itself.

Lastly, since movement is bidirectional, we need two ribbons between every pair of arc elements. Thus, we highlight the ribbon direction by 1) gradually changing the color along the ribbon from its source to destination but using the source color as the dominated color, and 2) putting a *ribbon gap* (see Figure 4) between the ribbon and its destination arc element. Hence, we can formulate an initial design of our interchange circos diagram as a visual representation of the interchange information at a junction, see Figure 4.

4.3. Improving Our Visual Design

However, this initial design still has a number of issues:

1. *Visual confusion.* Since the original circos figure treats all genome elements equally, it is thus natural to put the elements around the figure's circular border. Our case is, however, different because of a special link, i.e., the junction node itself. Hence, if we just present this link equally like the external connecting links, they can be mixed up, and potentially result in a visual confusion.
2. *Visual cluttering.* Second, for a junction node of valence n, we have $n(n-1)$ ribbons in total within an interchange

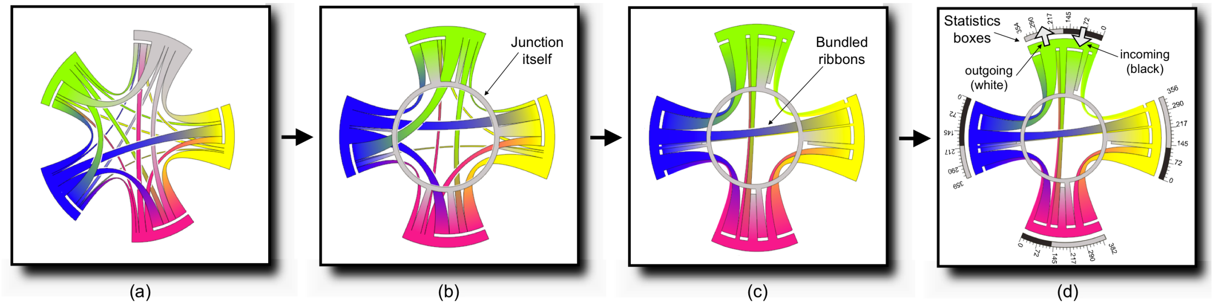


Figure 5: *Developing the interchange circos diagram from the original circos figure: (a) the initial design in Figure 4; (b) use a grey-colored flyover ring (like a source/sink) for the junction itself; (c) bundle pairs of bi-directional ribbons to reduce the visual cluttering; and (d) draw white and black curved statistics boxes to present the total outgoing and incoming flow volumes.*

circos diagram, e.g., the interchange circos diagram in Figure 5(a) has 20 ribbons. Even though we sort and render the ribbons, and apply haloes to enhance the visual occlusion, the intersecting ribbons could still be cluttered in spite of the fact that n is usually 4 or 5.

3. *Visual analysis.* Lastly, domain experts may want to directly observe basic statistics in the visual representation, but such information could still be missing, or not straightforward to be seen, e.g., comparing relative flow volume between bi-directional routes.

Hence we propose the following techniques to further improve our design:

1. *Flyover Ring.* To address the first issue above, we isolate the junction node, i.e., the source and sink of the interchange, from the other connecting links, and use a grey-colored flyover ring to represent the junction node, see Figure 5(b). In this way, we can avoid the visual confusion issue as well as reduce the number of ribbons.

2. *Bundling Ribbons.* To allow domain experts to visually compare the relative flow volumes between bidirectional ribbons between the same pair of links, we propose to bundle each pair of bidirectional ribbons together, see Figure 5(c). As for the labeled bundle shown in Figure 5(c), we can easily see that the blue-colored ribbon dominates; hence, there are far more people traveling from the blue to yellow link, than that of the opposite direction. In addition, this strategy can also help to address the visual cluttering problem by further reducing the total number of ribbons, e.g., from 20 in our initial design, to just the six ribbons shown in Figure 5(c).

3. *Statistics on Flow Volume.* Lastly, we draw a pair of black and white curved statistics boxes above each arc element with angular sizes proportional to the flow volumes along the corresponding link, see Figure 5(d). By these statistics boxes, one can quickly identify the relative flow volume along each link. Note that we use grey to indicate the outgoing flow and black for the incoming flow, and we may also optionally put in the actual numbers of the flow volume on the boxes.

4.4. Coloring Arc Elements

Since there are multiple interchange circos diagrams interconnected over the underlying traffic network, see Figure 1(e) or Figure 7, we propose to improve the visual connection between them by coloring their links (and the related arc elements) with the following two constraints:

- First, links connected to a common junction node should have different colors;
- Second, a common link between two neighboring junction nodes should have the same color.

This indeed is an edge coloring problem of an undirected graph, i.e., the traffic network G. Rather than using complex combinatorial optimization, since a junction node has at most seven links (which is a very rare case), we found that it is sufficient to fulfill the above two constraints by precomputing a small number of distinct colors and then applying a simple algorithm to assign these colors to the links:

Initialize:
for each edge in G **do**
  $c_{ij} = \emptyset$ $\triangleright$ c_{ij} is the link color between vertex i & j
end for
Main Loop:
for each edge in G (random order) **do**
  C_i = colors previously assigned to links of vertex i
  C_j = colors previously assigned to links of vertex j
  C = precomputed colors - $(C_i \cup C_j)$
  c_{ij} = randomly choose a color in C
end for

If k is the maximum vertex valency in G, the maximum number of neighboring links that any link would have is $2(k-1)$. Hence, precomputing $2k-1$ colors would be sufficient to fulfill the coloring constraints. In our implementation, we precompute a table of 13 colors ($k = 7$).

More than a single traffic network, we may have a series of traffic networks of different spatial scales. In this case, we should also attempt to maintain color coherence for links that exist in networks of consecutive spatial scales. This helps to maintain the visual context when one explores across spatial scales. To address this, we first apply our color assignment method to the coarsest-scale network graph, and then progressively color the links in the next finer-scale graph with an additional constraint:

- Third, if a link exists in two consecutive network graphs of different scales, we should try to assign a similar color to its two instances.

This is done by first checking if a link exists in the previous coarser graph and retrieving its color, say c_0, from the graph. If we need to enforce the third constraint, we assign c_0 to c_{ij} if c_0 is in C (see the main loop in pseudo code above), else we pick a color in C that is the most similar to c_0.

4.5. Positioning Arc Elements

When putting interchange circos diagrams that are geographically interconnected with one another, see again Figure 7, we have to scale and shift (angularly) the arc elements in each interchange circos diagram because of the following two issues. First, we need to scale the angular size of the arc elements, so that angular sizes can be used to indicate relative flow volume among links in the visualization. Second, taking the interchange circos diagram at the bottom of Figure 7(a) as an example, we need to shift the blue arc element, so that it roughly align with the direction toward the related interchange circos diagram on the right.

To address the first issue, we first determine the junction node that has the largest sum of in and out flow volumes in the current visualization view, e.g., the interchange circos diagram at the bottom of Figure 7(a). Then, we constrain the angle sum of all arc elements around it to be 180 degrees, and compute the angle size of every arc element in the visualization view by a simple linear proportionality based on its flow volume. By this simple idea, we can guarantee that the angle sum of arc elements around any node is no greater than 180 degrees, and that we can have sufficient angular space to shift the arc elements to resolve the second issue.

To further resolve the second issue to avoid overlapping the arc elements, every arc element in the current visualization is initially positioned in a way that it points toward its link direction. Then, in each interchange diagram, we simply check if any neighboring arc elements are too close to each other, and make them repel from each other. This is repeated iteratively until every pair of neighboring arc elements has a minimum gap of 10 degrees from each other.

4.6. Comparing with Existing Approach

Many existing visualization methods represent traffic flows by considering locations (junction nodes) in a pairwise manner. They aggregate the trajectory flows by computing only the total flow volume between every pair of neighboring nodes, and present these aggregated information as (bidirectional) arrows with varying width and color to show the corresponding flow volume. Such approach is intuitive and has been adopted in many applications, but it is not sufficient to reveal the interchange patterns because the interchange information has been lost when aggregating data.

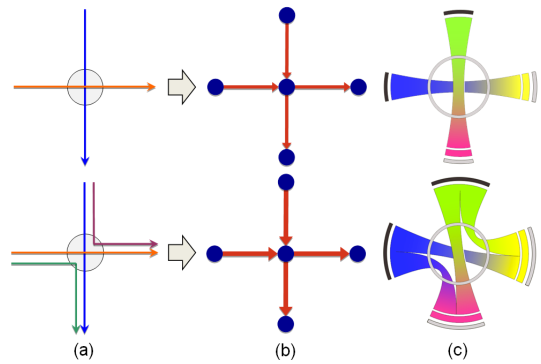


Figure 6: *Comparing interchange circos diagrams with existing visualization approach. (a) Two sets of raw trajectories; (b) Existing approach aggregates flows between pairs of locations and draws arrows to indicate the aggregated flow volume; (c) Our interchange circos diagrams are able to reveal the detail on the interchange patterns.*

Figure 6 compares interchange circos diagrams with the existing visualization approach. Here we show two simple examples that contain two and four trajectories, see top and bottom of Figure 6(a), respectively. After the data aggregation, both sets of trajectories result in a very similar aggregated visualization, see Figure 6(b), but in sharp contrast, our interchange circos diagrams are able to present to us clearly the difference in the interchange patterns emerged from the two trajectory sets, see Figure 6(c).

5. Interface: Visualizing Interchange

This section presents our visualization interface: 1) multi-scale visualization of interchange patterns, and 2) a family of interaction techniques for exploring interchange patterns.

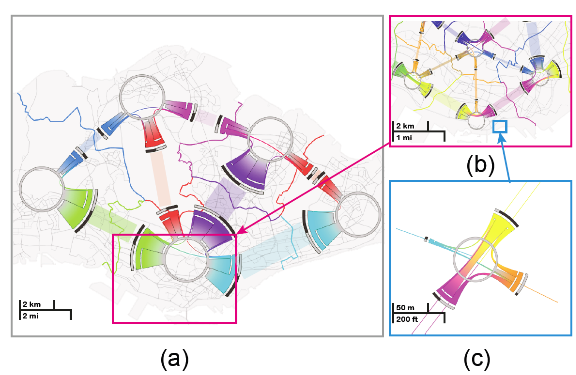


Figure 7: *Interchange Circos Diagrams in different scales: (a) city scale; (b) regional scale; and (c) road network scale.*

5.1. Multi-scale Visualization

As mentioned earlier, interchange patterns can emerge in different spatial scales, see Figure 7. Hence, given the traffic network graphs (of different scales) and the interchange matrices we precomputed from the raw trajectory data, see Section 3.2, we can plot the network graph associated with the current viewing scale in the visualization interface, and render the interchange circos diagrams at the visible junction nodes in that network graph.

Therefore, in case of the coarsest level (road level), we show one interchange circos diagram per road junction, and in case of region/city scales, where each partitioned area is a junction node, we show one interchange circos diagram per partitioned area and put it at the centroid of the area to avoid cluttering. See Figure 7 for the visualization results.

5.2. Interaction

Our system offers a family of interaction techniques to let users explore the interchange patterns.

- *Select.* The user can select an interchange circos diagram by click on it. After that, the related junction/region is highlighted as a visual feedback.
- *Zoom.* If a series of multi-scale traffic network graphs is available, the user can interactively zoom in/out to examine the interchange patterns in different spatial scale. In addition, our interface also provides an interactive magnifying glass function for users to do a focus+context visualization to examine the interchange patterns.
- *Roll.* In addition, one compelling feature of our interface is that the user can roll out a series of interchange circos diagram, see Figure 9, and observe the temporal changes of the interchange pattern over time.
- *Time Control.* Other than rolling to see temporal changes at a junction node, the user can also interactively adjust a timer control to filter the trajectory paths against a user-preferred time interval. By this, the user can animate all interchange circos diagrams in the visualization view and observe the temporal changes.

6. Implementation and Results

6.1. Movement Data

The data set consists of trajectories taken by over one million passengers who used the Singapore public transport, including metro and bus systems over a day. Each passenger carries her/his own RFID card, and uses it to tap in/out of the transportation carriers, e.g., buses and trains. Because of this, the public transport system can automatically record the entry and exit points of the passengers. However, since only starting and ending points of MRT (Mass Rapid Transit, Singapore) journeys are known, we assume the time shortest path as the passenger route so that we can obtain flows at the junctions [EFvE*12]. By this, trajectory paths can be extracted with time information for each passenger.

6.2. Implementation

Our system is implemented entirely in Java, so that it can run on different platforms in the future. Currently, it runs on an Intel Core i7 2 2GHz MacBook Pro with 8GB memory and an AMD Radeon HD 6490 graphics board.

Data storage. In the offline precomputation, see again Section 3.2, we mainly pre-compute interchange matrices for each junction node at all traffic network graphs over the partitioned time intervals. Note that we use 15 minutes as the time interval, so there are $24 \times 4 = 96$ partitioned time intervals over a day. Moreover, since there are about 1,600 junction nodes in total over all traffic network of different scales, and the interchange matrices are mostly 5×5 on average, the total memory needed to store the precomputed interchange data is around $96 \times 1600 \times 25 \times 4$ bytes, i.e. $\sim$15MB (note: we use 4-byte integers for the matrix elements).

Offline precomputation. Since it is impossible to load the entire raw trajectory data into the main memory, we divide the raw data into chunks and precompute the interchange matrices, i.e., $\sim$15MB data, for each chunk. Since interchange matrices of the same junction node can be summed, we can aggregate the overall interchange matrices for all raw trajectories by adding up matrices from the data chunks. It took about 30 minutes to preprocess one data chunk, and around 10 hours to finish the offline preprocessing.

Scalability of our Method. Since our visualization interface works with the precomputed interchange data, we do not need to load the raw trajectory data in the program runtime. Hence, it is independent of the amount of raw trajectories. However, it does depend on the time resolution we choose and the number of junctions we have in the traffic network graph because they affect the size of the precomputed interchange data.

6.3. Case study: Interchange at Metro Stations

Metro systems, usually referred to as subways or undergrounds, are massive and rapid public transportation crucial for the everyday life of many people around the world. Since metro systems serve like the backbone of metropolises with huge volume of daily traffic, having a visual interface to ex-

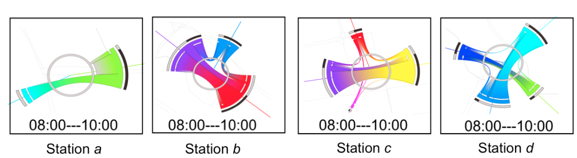


Figure 8: *Interchange patterns at four different train stations (a-d) in the Singapore Metro system.*

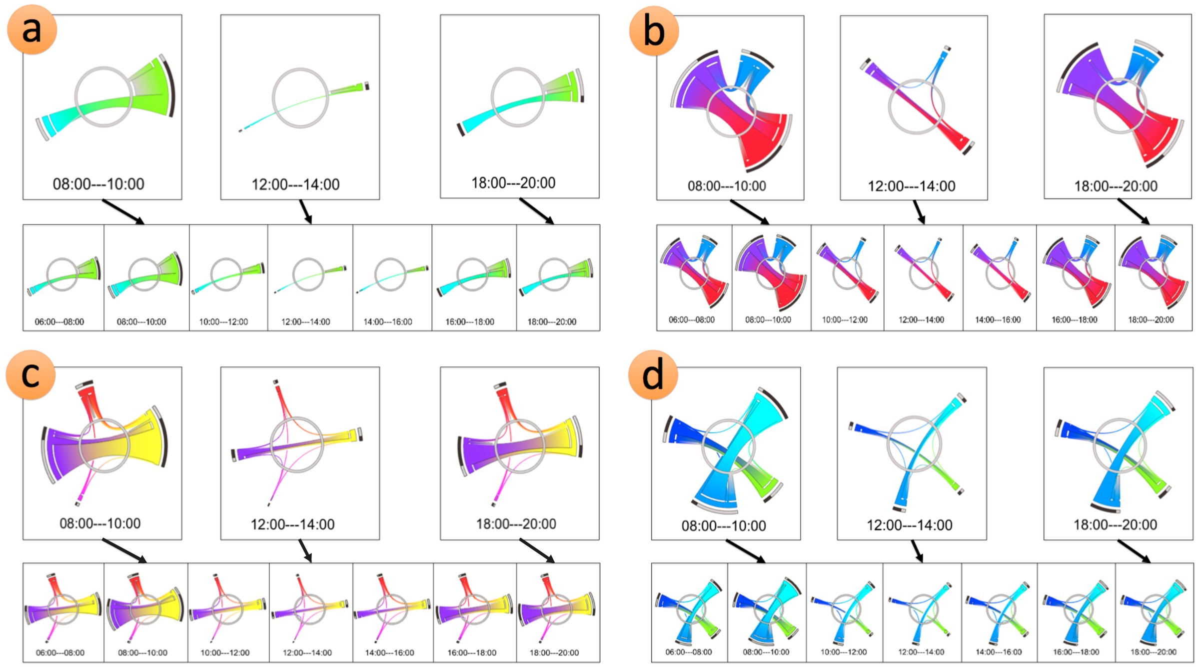


Figure 9: *Exploring the temporal changes (over a day) in the interchange patterns at four different train stations (a-d) in the Singapore Metro system.*

plore passenger interchange patterns at its train stations can help to examine how people distribute and move in a city.

Our system has been customized to analyze the interchange patterns at train stations in the Singapore Metro system, which now consists of four metro service lines and 89 train stations.

As shown in Figure 8, we can pick a train station and visualize its interchange pattern for a user-selected time interval, which is 8am to 10am in this case. By examining these four interchange circos diagrams, we can see the relative flow volumes for different possible routes at these train stations, e.g., the major movement directions at each station as well as the relative flow volumes among the four stations. Since the selected time interval is in the morning, we can observe unbalanced flow volumes in the bundled ribbons as well as in the node-connecting links.

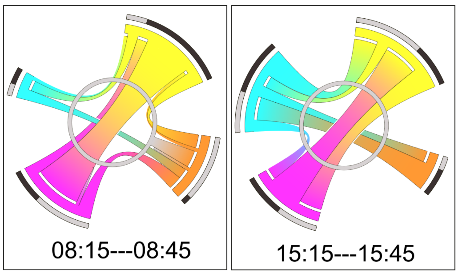
By using the "roll" operation, we can roll out a series of interchange circos diagrams, see Figure 9 to examine the temporal variations of interchange patterns at these four train stations. A common and general pattern shared by all four stations is that the most heavy traffic periods are the morning and evening peak hours, while there are far fewer passengers during the lunch hours. Specifically, we could observe the followings in our visualization:

- STATION *a* (Figure 9(a)) is the 2nd last station on MRT line 1. It only has two connecting links, and there are fewer traveling passengers compared to the other interchange stations as presented in the figure. In addition, our visualization can also reveal that during the morning peak hours, almost all passengers who enter this station come from the east side, and these people are almost equally distributed when they leave this station: roughly half of them continues their journey to the next station while the other half goes into the dominion of the station. And when evening comes, the interchange pattern basically reverses.
- STATION *b* (Figure 9(b)) is a busy interchange station connecting MRT lines 1 and 2. Comparing its temporal variation against that of the other three stations, we can clearly see that its flow volume is always larger than that of the others. Moreover, passengers who enter this station from line 2 on the top are (always) nearly equally re-distributed into the east and west connecting directions on line 1, and line 1 is usually busier than line 2. Lastly, we can also observe asymmetric flow volumes between the east and west connecting directions in this station during the morning and evening periods similar to that in STATION *a*.
- STATION *c* (Figure 9(c)) is an interchange station between MRT lines 1 and 4 with four connecting links. As seen from the figure, line 1 basically dominates the flow in this station. Though relatively fewer passengers on line 1 transit to line 4 here, passengers to or from line 4 appear to re-distribute fairly equally for all different outgoing routes in this station, showing that the newly-established line, i.e., MRT line 4, is like a supporting branch with line 1 being the main route.

- STATION *d* (Figure 9(d)) is an interchange station linking MRT lines 3 and 4. Interestingly, we find that the traffic flow volumes across the two lines are nearly the same, but these two service lines are relatively independent of each other, i.e., relatively not too many passengers transit between them, as compared to line 1 and 4 passengers in STATION *c*.

6.4. Case study: Intersection Capacity Utilization

The intersection capacity utilization (ICU) method [HA03] is a standard way in transportation research to measure the utilization rate of a road junction.



(a) (b)

Figure 10: *Comparison of lower (left) and higher (right) ICU ratings at a road junction during different time periods. (a) The traffic flow from yellow to violet dominates the junction utilization; moreover, both the orange and yellow connecting links are highly unbalanced. (b) Traffic flows from different links in the junction are fairly balanced and the incoming/outgoing flows for each connecting links are also fairly balanced.*

Our interface can also be used to estimate ICU at road junctions because one key factor that affects ICU is the relative amount of incoming and outgoing flow volumes from each direction at the road junction. Basically, the more balanced the flow volumes at different connecting links are, the junction will usually have a higher ICU rating.

The left and right hand sides of Figure 10 compare lower and higher ICU ratings, respectively, at a road junction during different time periods. Figure 10(a) has a lower ICU rating since the traffic flows from yellow to violet dominate the junction utilization; moreover, both the orange and yellow connecting links are highly unbalanced. Figure 10(b) has a higher ICU rating because traffic flows from each direction, as well as the incoming/outgoing flow volumes are relatively more balanced. With our interface, domain experts can efficiently identify potential road junctions with low ICU rating across different time of the day.

7. Conclusion

This paper presents a novel method of visualizing and exploring interchange patterns on real trajectory data of the Singapore public transportation system. First, we present a formal definition of interchange patterns, which is described as an interchange matrix that summarizes the flow volumes of different possible routes across a junction node. After that, we derive from the circos figure a new visual representation, *the interchange circos diagram*, to present the interchange information. Several practical issues to reduce visual cluttering and to improve the visual analytic capability have been considered to formulate this design, e.g., bundling bidirectional ribbons and statistics boxes to summarize flow volumes. Further than that, we also enhance the visual connection between neighboring diagrams and develop a working interface to present multiple interchange circos diagrams supported with a family of interaction operations. Lastly, we present two case studies to discuss how our interface was used to study the interchange patterns in the Singapore Metro system, and to examine the intersection capacity utilization (ICU) at junction nodes.

We have three future directions. First, we plan to design multitouch interaction for our interface. Second, we would like to experiment with transport data in other domains, e.g., network, web, and energy. Lastly, we plan to study visualization techniques related to biologic domain [BBG*09, NJBJ09] to try to further enhance our method, particularly about presenting flow directions and volumes.

Acknowledgment. This work was mainly established at the Singapore-ETH Centre for Global Environmental Sustainability (SEC), co-funded by the Singapore National Research Foundation (NRF) and ETH Zurich, and funded in part by the MOE Tier-1 fund (RG 29/11). Special thanks are due to Kay W. Axhausen and Alexander Erath of the Mobility and Transportation research module at the Future Cities Laboratory, M. Krzywinski for sharing the circos diagrams shown in Figure 3, and reviewers for the various constructive comments that help improve this manuscript.

References

[AA08] ANDRIENKO G., ANDRIENKO N.: Spatio-temporal aggregation for visual analysis of movements. In *IEEE Symp. on Visual Analytics Science and Tech.* (2008), pp. 51–58. 3

[AA11] ANDRIENKO N., ANDRIENKO G.: Spatial generalization and aggregation of massive movement data. *IEEE Trans. on Vis. and Comp. Graphics 17*, 2 (2011), 205–219. 1, 3

[AAD*08] ANDRIENKO G., ANDRIENKO N., DYKES J., FABRIKANT S., WACHOWICZ M.: Geovisualization of dynamics, movement and change: key issues and developing approaches in visualization research. *Info. Vis. 7*, 3 (2008), 173–180. 1

[AAPS08] ANDRIENKO N., ANDRIENKO G., PELEKIS N., SPACCAPIETRA S.: Basic concepts of movement data. In *Mobility, Data Mining and Privacy — Geographic Knowledge Discovery, Springer* (2008), pp. 15–38. 1

[AGMS11] ALSALLAKH B., GRÖLLER E., MIKSCH S., SUNTINGER M.: Contingency wheel: Visual analysis of large contingency tables. In *EuroVA 2011: International Workshop on Visual Analytics* (2011), The Eurographics Assoc., pp. 53–56. 4

[ARS79] APPEL A., ROHLF F., STEIN A.: The haloed line effect for hidden line elimination. In *SIGGRAPH* (1979), pp. 151–157. 4

[BBG*09] BLAAS J., BOTHA C., GRUNDY E., JONES M., LARAMEE R., POST F.: Smooth graphs for visual exploration of higher-order state transitions. *IEEE Trans. on Vis. and Comp. Graphics 15*, 6 (2009), 969–976. 9

[BOH11] BOSTOCK M., OGIEVETSKY V., HEER J.: D^3 data-driven documents. *IEEE Trans. on Vis. and Comp. Graphics 17*, 12 (2011), 2301–2309. 4

[DWL08] DODGE S., WEIBEL R., LAUTENSCHÜTZ A.-K.: Towards a taxonomy of movement patterns. *Info. Vis. 7*, 3 (2008), 240–252. 2

[EFvE*12] ERATH A., FOURIE P., VAN EGGERMOND M., ORDÓÑEZ S., CHAKIROV A., AXHAUSEN K.: Large-scale agent-based transport demand model for Singapore. In *International Conference on Travel Behaviour Research* (2012). 7

[Gan01] GANSKOPP D.: Manipulating cattle distribution with salt and water in large arid-land pastures: a GPS/GIS assessment. *Applied Animal Behaviour Science 73*, 4 (2001), 251–262. 1

[GNPP07] GIANNOTTI F., NANNI M., PINELLI F., PEDRESCHI D.: Trajectory pattern mining. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining* (2007), pp. 330–339. 1, 2

[Guo09] GUO D.: Flow mapping and multivariate visualization of large spatial interaction data. *IEEE Trans. on Vis. and Comp. Graphics 15*, 6 (2009), 1041–1048. 1, 3

[GvKS07] GUDMUNDSSON J., VAN KREVELD M., SPECKMANN B.: Efficient detection of patterns in 2D trajectories of moving points. *Geoinformatica 11*, 2 (2007), 195–215. 2

[GWY*11] GUO H., WANG Z., YU B., ZHAO H., YUAN X.: TripVista: Triple perspective visual trajectory analytics and its application on microscopic traffic data at a road intersection. In *Pacific Visualization Symp.* (2011), pp. 163–170. 2

[HA03] HUSCH D., ALBECK J.: *Intersection capacity utilization: Evaluation Procedures for Intersections and Interchanges*. Trafficware, 2003. 9

[HE02] HORNSBY K., EGENHOFER M.: Modeling moving objects over multiple granularities. *Annals of Mathematics and Artificial Intelligence 36*, 1–2 (2002), 177–194. 1

[HTC09] HURTER C., TISSOIRES B., CONVERSY S.: FromDaDy: Spreading aircraft trajectories across views to support iterative queries. *IEEE Trans. on Vis. and Comp. Graphics 15*, 6 (2009), 1017–1024. 2

[IG98] INTERRANTE V., GROSCH C.: Visualizing 3D flow. *IEEE Computer Graphics and Applications 18*, 4 (1998), 49–53. 4

[KSB*09] KRZYWINSKI M., SCHEIN J., BIROL İ., CONNORS J., GASCOYNE R., HORSMAN D., JONES S., MARRA M.: Circos: an information aesthetic for comparative genomics. *Genome research 19*, 9 (2009), 1639–1645. 2, 4

[KW04] KAPLER T., WRIGHT W.: GeoTime information visualization. In *Information Visualization* (2004), pp. 25–32. 1, 2

[Kwa00] KWAN M.-P.: Interactive geovisualization of activity-travel patterns using three-dimensional geographical information systems: a methodological exploration with a large data set. *Transportation Research Part C: Emerging Technologies 8*, 1 (2000), 185–203. 1, 2

[LGL*11] LIU H., GAO Y., LU L., LIU S., QU H., NI L.: Visual analysis of route diversity. In *IEEE Conference on Visual Analytics Science and Tech.* (2011), pp. 171–180. 2

[LKI05] LAUBE P., KREVELD M., IMFELD S.: Finding REMO — detecting relative motion patterns in geospatial lifelines. *Developments in spatial data handling* (2005), 201–215. 1, 2

[MK01] MACEACHREN A., KRAAK M.: Research challenges in geovisualization. *Cartography and Geographic Information Science 28*, 1 (2001), 3–12. 2

[NJBJ09] NIELSEN C. B., JACKMAN S. D., BIROL I., JONES S. J.: ABySS-Explorer: visualizing genome sequence assemblies. *IEEE Trans. on Vis. and Comp. Graphics 15*, 6 (2009), 881–888. 9

[PKM*07] PELEKIS N., KOPANAKIS I., MARKETOS G., NTOUTSI I., ANDRIENKO G., THEODORIDIS Y.: Similarity search in trajectory databases. In *14th International Symp. on Temporal Representation and Reasoning* (2007), pp. 129–140. 2

[RPN*08] RINZIVILLO S., PEDRESCHI D., NANNI M., GIANNOTTI F., ANDRIENKO N., ANDRIENKO G.: Visually driven analysis of movement data by progressive clustering. *Info. Vis. 7*, 3 (2008), 225–239. 2

[SvLM11] SEWALL J., VAN DEN BERG J., LIN M. C., MANOCHA D.: Virtualized traffic: Reconstructing traffic flows from discrete spatio-temporal data. *IEEE Trans. on Vis. and Comp. Graphics 17*, 1 (2011), 26–37. 1

[SWvdW*11] SCHEEPENS R., WILLEMS N., VAN DE WETERING H., ANDRIENKO G., ANDRIENKO N., VAN WIJK J.: Composite density maps for multivariate trajectories. *IEEE Trans. on Vis. and Comp. Graphics 17*, 12 (2011), 2518–2527. 2

[SWvdWvW11] SCHEEPENS R., WILLEMS N., VAN DE WETERING H., VAN WIJK J.: Interactive visualization of multivariate trajectory data with density maps. In *Pacific Visualization Symp.* (2011), pp. 147–154. 2

[TSAA12] TOMINSKI C., SCHUMANN H., ANDRIENKO G. L., ANDRIENKO N. V.: Stacking-based visualization of trajectory attribute data. *IEEE Trans. on Vis. and Comp. Graphics 18*, 12 (2012), 2565–2574. 3

[WDSC07] WOOD J., DYKES J., SLINGSBY A., CLARKE K.: Interactive visual exploration of a large spatio-temporal dataset: reflections on a geovisualization mashup. *IEEE Trans. on Vis. and Comp. Graphics 13*, 6 (2007), 1176–1183. 2

[WSL12] WILKIE D., SEWALL J., LIN M. C.: Transforming GIS data into functional road models for large-scale traffic simulation. *IEEE Trans. on Vis. and Comp. Graphics 18*, 6 (2012), 890–901. 1

[WvdWvW09] WILLEMS N., VAN DE WETERING H., VAN WIJK J.: Visualization of vessel movements. In *Computer Graphics Forum* (2009), vol. 28, pp. 959–966. 2

[ZZA*12] ZENG W., ZHONG C., ANWAR A., MÜLLER ARISONA S., MCLOUGHLIN I.: MetroBuzz: Interactive 3D visualization of spatiotemporal data. In *Computer & Information Science (ICCIS), 2012 International Conference on* (2012), vol. 1, IEEE, pp. 143–147. 2