MovementSlicer: Better Gantt Charts for Visualizing Behaviors and Meetings in Movement Data

Shrey Gupta* ÉTS
Maxime Dumas† ÉTS
Michael J. McGuffin‡ ÉTS
Thomas Kapler§ Uncharted Software Inc.

ABSTRACT

Movement data collected through GPS or other technologies is increasingly common, but is difficult to visualize due to overplotting and occlusion of movements when displayed on 2D maps. An additional challenge is the extraction of useful higher-level information (such as meetings) derived from the raw movement data. We present a design study of MovementSlicer, a tool for visualizing the places visited, and behaviors of, individual actors, and also the meetings between multiple actors. We first present a taxonomy of visualizations of movement data, and then consider tasks to support when analyzing movement data and especially meetings of multiple actors. We argue that Gantt charts have many advantages for understanding the movements and meetings of small groups of moving entities, and present the design of a Gantt chart that can nest people within locations or locations within people along the vertical axis, and show time along the horizontal axis. The rows of our Gantt chart are sorted by activity level and can be filtered using a weighted adjacency matrix showing meetings between people. Empty time intervals in the Gantt chart can be automatically folded, with smoothly animated transitions, yielding a multi-focal view. Case studies demonstrate the utility of our prototype.

Index Terms: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

1 INTRODUCTION

GPS and other position-tracking technologies yield increasingly common and large movement datasets (sequences of positions over time). Visualizing such data is challenging for a few reasons. First, multiple variables are involved: *latitude* and *longitude* as a function of *time* and *object id*, where object id identifies a person, vehicle, or other moving entity. Second, the movements often cross each other and may repeatedly travel other the same pathways to the same locations, causing occlusion (overplotting) when drawn on a 2D geographic map. Third, movements may occur at widely varying physical scales, e.g., ranging from tens of meters (moving between two buildings) to hundreds of kilometers (traveling between cities) in the case of a single GPS dataset, and zooming out on a 2D map will leave only the largest-scale movements salient. Fourth, the data may cover long spans of time containing thousands of events.

Many previous approaches to visualizing movement data have involved the display or analysis of the *shapes* of movement trajectories [25, 13, 22, 19], or have proposed ways of dealing with large numbers of moving objects [48, 8, 3, 47, 51, 5], or both [30, 9, 27, 49], often making use of aggregation [14].

In the current work, we are instead interested in understanding a small number (< 20) of moving objects. In such a case, we would like to avoid aggregating groups of objects, so that the history of each entity is visible. At the same time, we are *not* interested in the detailed shape of movement trajectories. Instead, we are interested in the *discrete locations* (e.g., rooms, workstations in a factory, buildings, addresses, cities) visited by the moving entities. We are interested in *where* the people or objects have been (in terms of discrete locations), *when* they were there, *how many times* they visited different locations, and *in what order*. We are also interested in *meetings* (i.e., same place and time) that occur between the objects or persons. Such information is difficult to convey in a single 2D geographic map, due to overplotting or the need for animation.

There are several scenarios where we may wish to understand the movements of a small number of objects or persons over a set of discrete locations. These include analyzing meetings and activities of suspected criminals (gang members, terrorists) whose cell phones are tracked; monitoring offenders on probation; understanding movements of a team of workers and their equipment in a factory, to improve workflow processes; analyzing movements of visitors in a museum; understanding movements of health care professionals, patients, and equipment within a hospital, to optimize changes to floorplans or reduce the risk of pathogens spreading; or movements of several customers within a large store, to optimize merchandise displays and layout. Such scenarios may also arise if the user begins with an overview of a large, aggregated dataset, and selects a small subset of moving entities to analyze in more detail.

In these situations with a small number of moving objects and no need to understand the shapes of trajectories, we can discretize space to simplify the visualization. Such discretizations are leveraged in previous work [3, 43, 46, 5] to simplify 2D geographic maps, but these approaches either require aggregation of sets of moving entities, or else may still suffer from overplotting. In the current work, we instead *list* the discretized locations as rows in a kind of Gantt chart. In the past, Gantt charts have been used for schedules of activities [17, 37, 44, 18, 24], software execution threads [45, 23], and dynamic graphs [21], but Gantt charts have received little attention for visualizing movement data (the only example we know of is Andrienko et al. [6], who use the vertical axis in a Gantt chart to list different moving objects, but not to list locations). Compared to other visualizations of movement data, Gantt charts have the advantage of offering a single, 2D view, free of occlusion, displaying the data without the need to aggregate groups of moving objects, and easily handle movements between different locations at widely different scales. We demonstrate that Gantt charts support several analytical tasks for movement data, and contribute a design study that goes beyond previous literature by identifying multiple ways to display *both* locations *and* moving entities within a Gantt chart, as well as ways to highlight meetings of moving entities. We also apply techniques to make Gantt charts more scalable vertically (by allowing rows to be filtered and sorted) and horizontally (by allowing the user to fold empty spans of time).

Our contributions are (1) a simple taxonomy of ways of displaying movement data; (2) the identification of tasks relevant for understanding the movements and meetings of small groups of entities; (3) a discussion of design choices around Gantt charts for movement data, analyzing ways to show two variables (person and location) as a function of time, and ways to show meetings; (4)

*e-mail: shrey.gupta.1@ens.etsmtl.ca
†e-mail: maxime.dumas.1@ens.etsmtl.ca
‡e-mail: michael.mcguffin@etsmtl.ca
§e-mail: tkapler@uncharted.software

a prototype with several features for visualizing movement data; and (5) case studies to illustrate the use of the prototype. Our prototype uses color, and an explicitly drawn link, and animation to link together multiple coordinated views. It also allows for filtering through an adjacency matrix of meetings between people; has a Gantt chart that can show people within places or places within people as a function of time; allows sorting, filtering, and annotation of rows in the Gantt chart; enables multi-focal "folding" to allow the user to see details of meetings and events with intervening empty time periods compressed; and uses smoothly animated transitions when folding or unfolding time.

2 Related Work

Andrienko and Andrienko provide an overview [4] of techniques for visualization and analysis of movement data. From a visual perspective, without considering data aggregation, we note that many visualizations of movements fall into 5 categories (Figure 1).

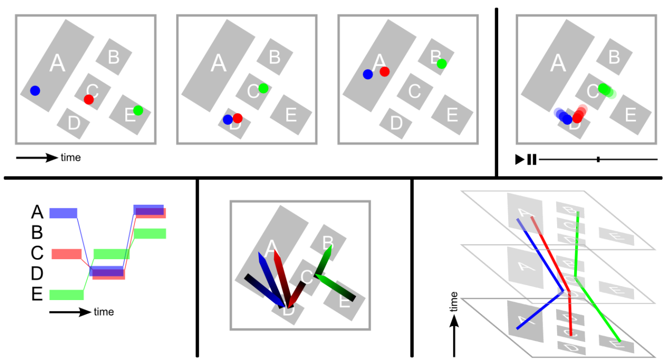

Figure 1: A taxonomy of visual representations of the same movement data. *Top left:* small multiples. *Top right:* animation with playback controls. *Bottom left:* Gantt chart. *Bottom middle:* static 2D geographic map with trajectories. *Bottom right:* 3D "space-time cube".

The 3D view (Figure 1, lower right) is sometimes called a "space-time cube" [20, 28, 26, 7, 1] and has been used in a successful commercial product [25]. Such views require 3D camera controls to rotate the visualization to different points of view.

The small multiples in Figure 1 have the drawback of requiring more space (or must be shrunk down in size, reducing the spatial resolution of each snap shot). Animated playback, on the other hand, requires time to watch. And 2D geographic maps (Figure 1, lower center) suffer from overplotting and do not show timing.

The Gantt chart (Figure 1, lower left) transforms position (latitude and longitude) into a single (vertical) axis. This has two disadvantages: the user may no longer judge the geographic distances between locations, nor see the shape of trajectories taken between locations. However, there are advantages of this approach: (1) there is no longer any occlusion due to trajectories crossing each other or revisiting the same locations multiple times; (2) it is clear in what order and how many times each location is visited; (3) locations can be sorted vertically according to various criteria, such as frequency of activity, to make it easier to find commonly (or uncommonly) visited locations; (4) movements at different scales (e.g., between two neighboring buildings, and later between two cities) in the same dataset pose no difficulty and are simultaneously visible. The simplicity of the Gantt chart allows many questions related to time and space to be answered, as we will see later.

Gantt charts show at least one categorical (or ordinal) variable crossed with time, and were originally proposed for visualizing work schedules [17]. They are still used for schedules in recent research [44, 24], and for showing intervals over time in other contexts [37, 6]. The Gantt chart that we propose, unlike previous work on movement data, maps *two* variables to the vertical axis (location and person), and we discuss the tradeoffs of nesting location inside person, or vice versa, during this mapping.

Crnovrsanin et al. [13] propose an approach somewhat related to the Gantt chart, that maps a continuous *distance* to the vertical axis, which varies with time shown on the horizontal axis. The distance shown is computed with respect to some fixed (or moving) point of reference. This way, objects that move toward or away from each other geographically also usually move toward or away from each other in Crnovrsanin et al.'s "distance $\times$ time" chart. This allows more details of trajectories to be seen. In our work, we choose a Gantt chart so as to emphasize the locations where people are stationary, and to avoid the problems with widely-different scales of motion mentioned in the introduction.

Tanahashi and Ma [42] present ways to improve the layout of "storyline visualizations", which are somewhat similar to Gantt charts, but are usually used to show the temporal *ordering* of events without showing their actual durations, and also do not normally have any meaningful vertical axis.

Guo et al. [19] interpret trajectories as multivariate data (e.g., computing each trajectory's maximum speed, average speed, angular range, etc.) and display them as tuples within a parallel coordinates plot that is also coordinated with other views. Other recent work [48, 8, 3, 30, 47, 51, 27, 5] has proposed ways of aggregating movement data to show overviews. Again, our work is not geared toward overviews of large numbers of objects, nor towards visualizing trajectories, but rather understanding behaviors of small groups of moving people or objects.

3 Meeting-Oriented Tasks

Andrienko et al. [2] propose a general framework for tasks surrounding movement data, and Amini et al. [1] propose a simpler taxonomy of tasks. We present here an even simpler, but still useful, set of categories of tasks that are relevant for understanding the activities of small groups of moving entities.

Person-centric: These questions concern a particular person, e.g., "Where was Alice at a particular time?", "What are the most commonly visited places by Bob, and in what order are they usually visited?", and "What are some rarely visited places by Carol?"

Location-centric: These concern a particular place, e.g., "Who was at the store on a particular day?", "Who most often visits the office building?", and "Who rarely visits the back alley?"

Meeting-centric: Meetings, unlike people, have a limited duration in addition to a location. Questions about them include "Who are the groups of people who meet together?", "How many times, when, and where did they meet?", "How many times, and when, did a group meet at a particular place?", or "Who is usually earliest, or latest, for a group meeting?"

These categories are not mutually exclusive: the question "When was Dave at home?" is both person-centric and location-centric.

In considering design options, we can consider how well the above categories of questions are supported.

4 Gantt Charts

4.1 Why Gantt Charts?

The main alternatives to Gantt charts are shown in Figure 1. These alternatives suffer from occlusion, require interaction to see more than a single moment in time, or (in the case of small multiples) require much mental integration to notice any patterns as well as suffer from reduced spatial resolution. Also, none of the alternatives to Gantt charts can clearly show movements on widely varying spatial scales, such as movements of a few meters and also several kilometers, on the same map, because all of the alternatives are stuck with using a geographic map in some way in their presentation.

Gantt charts eliminate the overplotting that would occur on 2D geographic maps, and also solve the problem of widely-varying scales of distance between locations of interest. This eliminates two of the four problems mentioned in the first paragraph of the introduction. Gantt charts also make it possible to see three variables at once (time, location, and object id), and can be made to scale better along time via folding (discussed later), which at least mitigates the other problems mentioned in our first paragraph.

Gantt charts make several behaviors visible (Figure 2), allowing us to see frequently and infrequently visited places, repetitions and outliers, and changes in patterns.

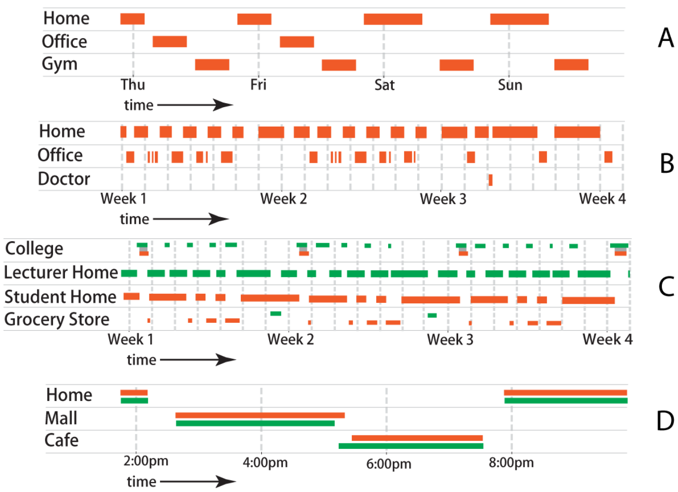

Figure 2: Patterns, outliers, meetings, and parallel movements. *A* shows someone going to work and the gym each weekday, and then only to the gym on the weekend. *B* shows a change in pattern: after going to work for 2 weeks, this person visits a doctor, then stays home several days (perhaps due to sickness). *C* illustrates possible meetings: a student and lecturer sometime coincide on the campus of a college, while also visiting a grocery store at different times. *D* shows movement together: two people leave home to visit the mall. Around 5:30pm, one goes to a café before being joined by the other a bit later, then both return home.

Notice that, in the charts of Figure 2, every location has a dedicated row, whereas the identity of people is shown by color. Use of color in this way does not scale well beyond ≈10 people. Also, where there are many people and/or many locations, a single person may "jump" vertically by arbitrary distances. This could make it difficult to answer person-centric questions, as it can be difficult for the user's eyes to trace the activity of a single person. The next section shows that there are multiple ways to design Gantt charts, some of which address these shortcomings.

4.2 Design Choices

As mentioned in the introduction, we can model movement data with 4 variables: *latitude* and *longitude*, that are functions of *time* and *object id*. For convenience in our discussion, we will use *person* as a synonym of *object id*.

Because we are interested primarily in the discrete locations that a person visits (such as home, work, store, café, ...), we can replace *latitude* and *longitude* with a single *location* variable. We can then map *time* to a horizontal axis, and use dimensional stacking [29, 34] to map both *location* and *person* to the vertical axis. Depending on the order of nesting of variables, we end up with a chart resembling Figure 3 A or B.

Figures 3 A and C are person-centric, whereas B, D, and E are location-centric. Each of these is more suited to either the person-

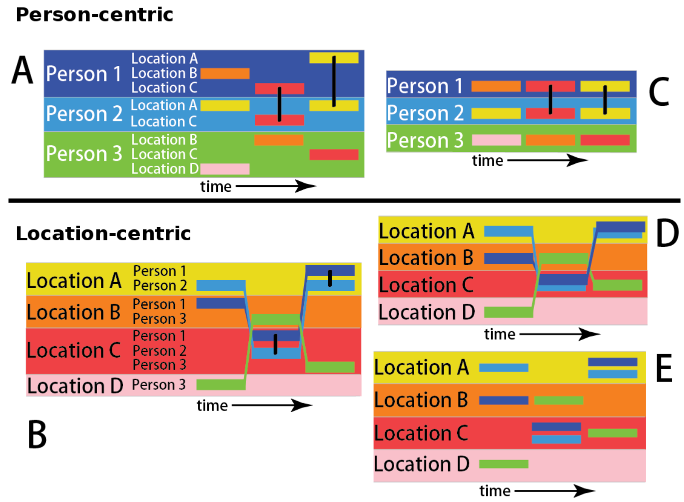

Figure 3: All variants of these Gantt charts show the same fictitious data, and all use the same warm colors for locations and cold colors for persons. *A* is person-centric, and *B* is location-centric, each of them nesting one variable within another along the vertical axis. C, D, and E show ways of making rows more narrow. C does this by exploiting the fact that a person can only be in one place at a time. D and E either overlap or stack people, respectively, when multiple people are in the same location. In all charts, the background colors of rows are superfluous and only for illustration. Furthermore, the foreground colors used in A and B are not strictly necessary, though they may help with visual search tasks. Vertical black line segments are used to highlight meetings between people.

centric or location-centric questions presented in Section 3. For example, to answer the location-centric question "Who most often visits the store?", in a location-centric chart, the user simply needs to look across the row for the store, and examine the contents of that row. Answering the same question in a person-centric chart would requiring looking over many or all rows.

Thus, there is good reason to support both person-centric and location-centric visualizations, to facilitate both kinds of questions.

Of the two person-centric charts sketched in Figure 3 (A and C), C has the advantage of saving space vertically, but requires that locations be color-coded. The datasets we have worked with contain many more locations than people, and therefore color coding of locations would not scale well. Figure 3 A, however, does not *require* any color coding. Therefore, **we choose A** as one of the charts we implemented.

Of the three location-centric charts (B, D, E), **we choose E** because it saves space vertically over B, and does not depend on overplotting as in D. E may or may not require people to be color-coded for clarity, depending on the way it is implemented. (In our prototype implementation, our location-centric view's color-coding of people is *optional but not necessary*, as people can also be identified by a tooltip or excentric labels [16] and also through coordinated highlighting with another view listing people's names.)

4.2.1 Depiction of Meetings

Meetings are clearly illustrated in location-centric views. We suspect that location-centric views will be better for visualizing meetings of people, to show the activity of the whole group within one localized region of the chart.

However, in a *person*-centric view, it was not initially obvious how to best highlight meetings, since each location may appear as multiple, non-contiguous rows or multiple positions on the vertical axis. We first tried drawing connections between the centers of

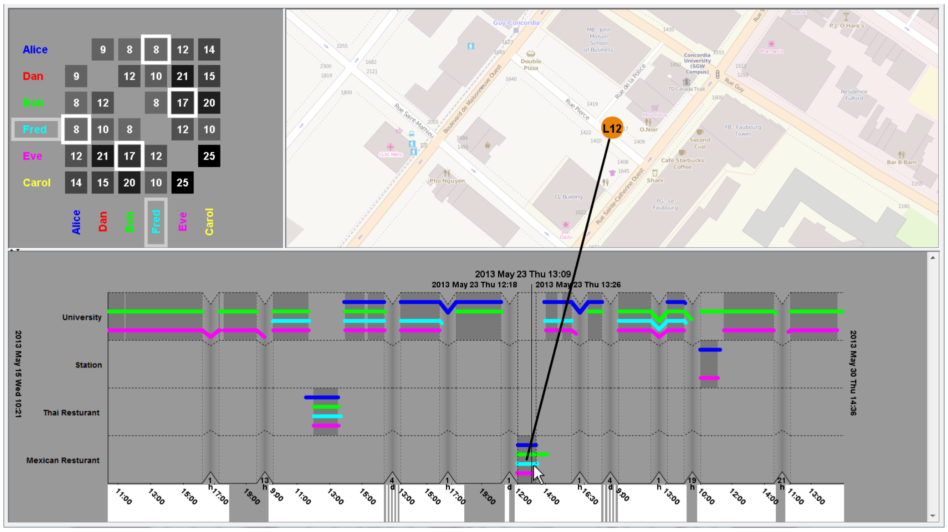

Figure 4: The main window. Upper right: 2D geographic map. Upper left: adjacency matrix showing meetings between people. Bottom: Gantt chart, in location-centric mode, with time folding activated to hide empty regions of time. Note the small labels below each fold indicated how much time has been compressed ("13 h" for 13 hours, "4 d" for 4 days, etc.).

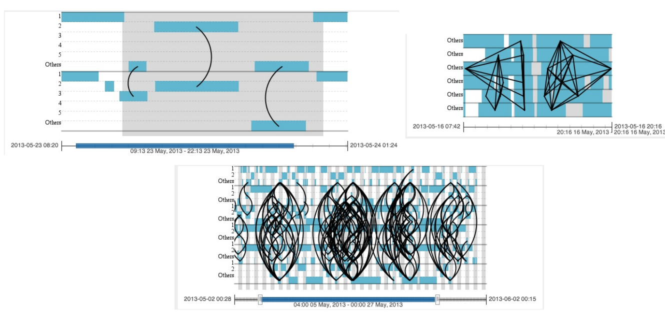

Figure 5: Early, unsatisfactory attempts to indicate meetings in person-centric views, drawing either curves or straight line segments between the centers of intervals of all participating people.

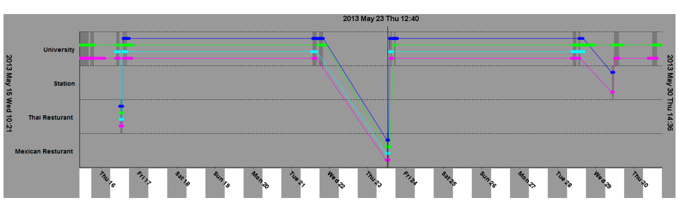

Figure 6: The same time span as in Figure 4, but now with folds removed, to show how much empty space there is between meetings. World lines are also displayed.

all pairs of relevant intervals, and tried both curved and straight connections (Figure 5). This yielded excessively complex results when a meeting involved several people. An alternative approach is to draw a single vertical line segment for each meeting, connecting all relevant intervals, however this may create ambiguities if the line segment happens to pass over intervals of people *not* participating in the meeting. Our final solution was to add a round dot to indicate which people were truly participating in the meeting (Figure 8).

4.2.2 World Lines in Location-Centric Views

In location-centric views, another design issue is whether to show "world lines", i.e., polygonal line segments connecting each person across the locations they visit. This is shown in Figure 3 B and D, but not in E. We recommend making these lines optional. Displaying them can make each person's path more salient (Figures 6 and 10), but can also create visual noise if the user is zoomed out in time and there are many nearly vertical line segments connecting the locations of a person.

5 MOVEMENTSLICER PROTOTYPE

5.1 Features

MovementSlicer (Figure 4) was implemented in Java, using Swing widgets and the JXMapViewer Swing component for rendering geographic maps with cached tiles that can be retrieved locally or over a network. A session begins with the user choosing movement data files to read in, which are processed by a clustering algorithm we implemented in Java, based on [52]. The clustering algorithm identifies locations where people have remained within some maximum radius of space for some minimum amount of time. Meetings are then identified automatically by checking which locations were visited by different people during overlapping time intervals.

An adjacency matrix, with each person assigned to a row and column, shows the number of meetings between each pair of people. The matrix is ordered using the barycenter heuristic [41, 33], to cluster subsets of people who meet with each other. This is important to make it easier to see groups of people who interact frequently. The matrix also serves as a filtering mechanism. The user may select one or more people in the matrix, causing only data for those people to appear in the other views. The user may also select one or more cells in the matrix, causing only data related to *meetings* of people to appear in the other views. Thus, although 10 initial people might result in a large number of rows in the Gantt chart, selecting only the cells (meetings) of these people in the matrix typically greatly reduces the number of rows in the Gantt chart.

The views in the main window are coordinated [36, 50, 39] in various ways. First, hovering over a location in the Gantt chart causes the geographic map to animate to that location (coordination through animation), and the relationship is further emphasized with a black line segment drawn from the cursor in the Gantt chart to the

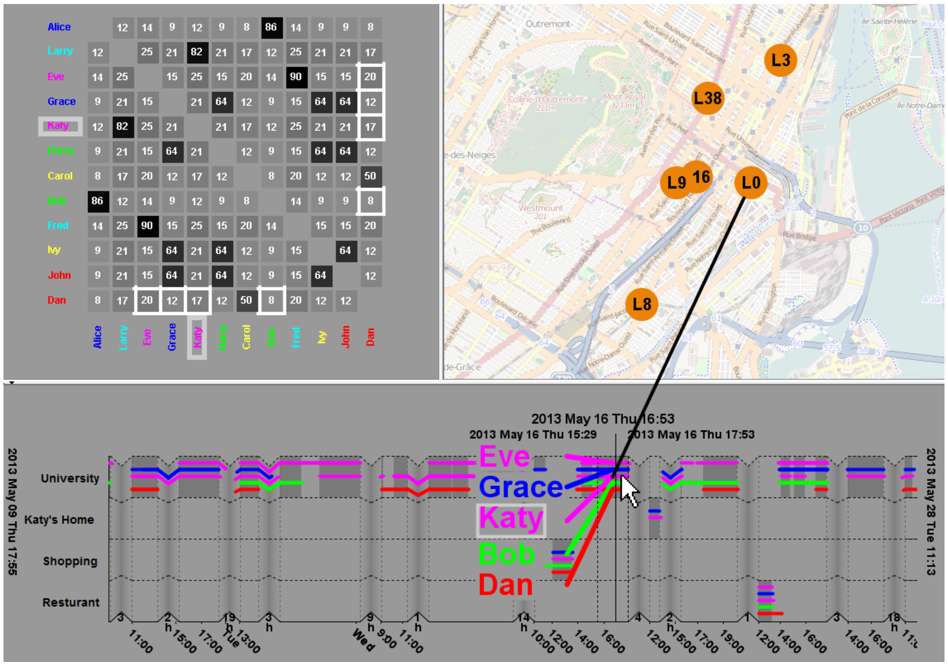

Figure 7: From an initial set of 12 people shown in the matrix, the user has selected 6 people, causing their meetings to be displayed in the Gantt chart. The identity of people in the Gantt chart is revealed in three ways: through coordinated highlighting ("Katy" is under the mouse cursor, and her name is highlighted with a grey rectangle in the matrix view), through excentric labels [16], and through color coding.

location in the map (such explicit links have also been discussed [10] and used [12, 40, 43] in previous work).

To color-code people, we carefully chose a set of 6 foreground colors and a background shade of grey for maximal contrast (this coding is optional but not necessary). Colors are recycled when there are more than 6 people, in which case a tooltip, excentric labels, or highlighting enables disambiguation.

The Gantt chart in Figure 4 is location-centric. The user has selected cells in the matrix to view meetings between Fred, Alice, Eve, and Bob. Meetings are shown in the Gantt chart as darker grey bands, and the horizontal colored strokes allow the user to see when each person arrived and left the meeting, and see the duration of each person's stay. It is straightforward to see who arrived first or last at a meeting, or who left first or last, because the temporal folds increase the horizontal resolution available for each meeting, and the people in the same meeting appear vertically very close.

The temporal folding supported by the Gantt chart is a focus-in-context technique [11], comparable to other work where folds are performed along one dimension [32, 15, 45]. We found the darker shading and converging lines in our folds were important to convey a metaphor of receding planar surfaces.

Figure 6 shows the same data, over the same time span, without folds. Notice that the empty space between meetings has greatly reduced the horizontal resolution available for meetings, making it difficult to see details such as who arrived first at a meeting.

In both Figures 4 and 6, a black-and-grey strip pattern below the Gantt chart shows day and night (white for day, grey for night). Also, numbers below folds in Figure 4 indicate the time hidden by each fold: "13 h" for 13 hours, or "4 d" for 4 days.

The Gantt chart and the geographic map can both be panned and zoomed by the user. Hotkeys are used to toggle folding in the Gantt chart, as well as to toggle the drawing of world lines. During the toggling of folding, a smoothly animated transition shows the empty regions folding away, while the unfolded regions grow bigger, maintaining the same total duration within the Gantt chart.

Names of locations in the left margin of the Gantt chart can be clicked and edited by the user, to annotate rows with meaningful location names. Thus, during an analytical process, as the user identifies locations on a map, they can record a meaningful name for each location, such as "Building 1", "Campus", "Bob's Home", etc.

As the cursor rolls over different people in the Gantt chart, the corresponding person's name is highlighted in the matrix view with a grey rectangle ("Fred" in Figure 4, "Katy" in Figure 7). We may also optionally activate a tooltip (not shown) that displays the person's name beside the cursor, or activate excentric labels [16] that display all the person's names (Figure 7) with the person under the cursor highlighted with a grey rectangle in the Gantt chart.

The Gantt chart supports both the location-centric view shown in Figures 4, 6, 7, and the person-centric view shown in Figure 8. The user may freely switch between these views according to the kinds of tasks or questions they have.

When using the location-centric view, rows within the Gantt chart are sorted by default in descending order of activity, with the most visited locations at the top. The user may also reverse this sorting order. This makes it easy for the user to identify the most-often visited locations, or the most rarely visited locations, for a given set of people selected in the adjacency matrix. The user may also interactively adjust the height of rows, and scroll vertically through the Gantt chart if the number of rows exceeds the available space.

To summarize, although the number of rows that are visible in the Gantt chart is limited by screen dimensions, we increase the scalability of our system in multiple ways: the adjacency matrix allows the user to identify interesting meetings and filter on people, sorting of rows makes the most interesting (frequent or rare) locations appear at the top of the Gantt chart, and folding increases scalability along time and reduces the need to pan and zoom. The user may also remove individual locations by right-clicking on rows.

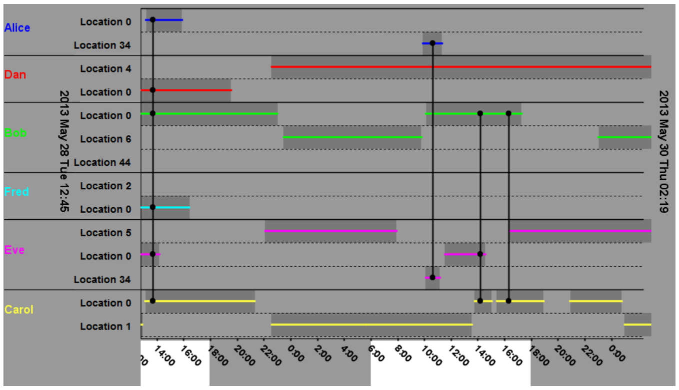

Figure 8: Person-centric mode, showing four meetings. People are identified by their vertical position and also (redundantly) by their color (e.g., Alice is in the *top* row, and is blue). Black vertical line segments show meetings, with black dots disambiguating the participants in each meetings. For example, the 1st meeting involves all 6 people at Location 0, but the 2nd meeting only involves Alice and Eva at Location 34.

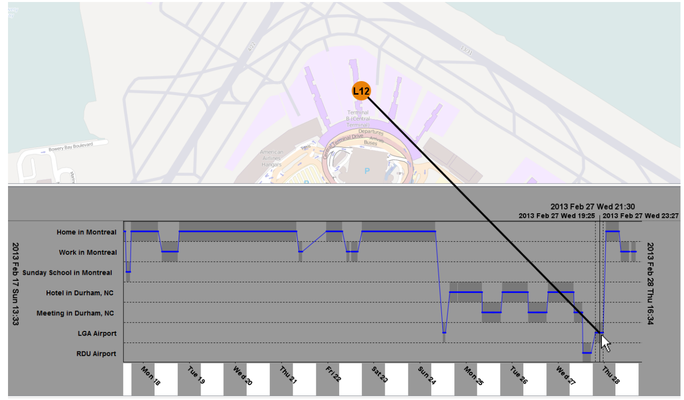

Figure 10: Case study 1: one week of activity in Montreal followed by a round-trip flight to Durham (airport code: RDU) via New York (airport code: LGA). GPS tracking failed during the LGA-RDU leg of the trip, which shows up in the Gantt chart as a sudden jump from LGA to a hotel in Durham.

6 Case Studies

6.1 Case Study 1: One individual over 6 months

This dataset was produced by tracking one person's smartphone for 6 months, yielding a total of 21,530 data points. In Figure 9, we see 3 months of data, all within the same city. Although there are initially 41 locations identified in the data, and 41 rows displayed, the rows are automatically ordered in descending order of activity, allowing the user to easily see the most frequently visited locations without scrolling downward. The user manually inspected each row to see the corresponding location on the map, and annotated the rows in the Gantt chart ("Home", "Work", "Strip Mall", etc.) to make the visualization more suitable for interpretation and presentations to others.

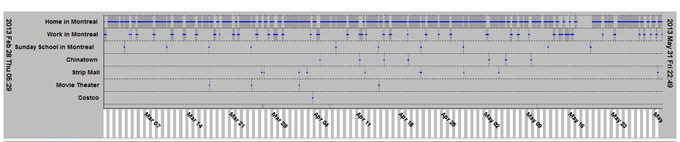

Figure 9: Case study 1: three months of activity within Montreal.

In Figure 10, we see a trip from Montreal, Canada to Durham, NC, within the same dataset. The data covers a large range of geographic distances: flight distance from Montreal to Durham covers over 1000 km, but travelling between "Hotel in Durham" and "Meeting in Durham" covers less than 3 kilometers. Zooming out on a geographic map to see all places visited, we would only see the 3 cities covered: Montreal, Durham, and New York City. The details of movements within any city are far too small to see in such a geographic overview. In contrast, with the Gantt chart, all locations where the person visited are allocated a row, making the sequencing and timing salient. This demonstrates an important advantage of the Gantt chart over the other approaches in Figure 1.

6.2 Case Study 2: Six people over 1 month

Six people in our research team tracked their movements over a 1 month period, while traveling to and from work, and occasionally meeting in places outside work such as restaurants. Our implementation is able to read in the data for all six people (135k raw points, or roughly 1 point per person every 2 minutes), containing gaps and noise, and process and cluster it to find visited locations in under 5 seconds on a recent laptop. Our implementation also identifies meetings between multiple people. Visualizing such meetings is important for answering questions such as "Who among these people often meet together?", "Where do they meet?", and "Who is early or late for a meeting?"

Initially, 78 different locations are identified in the dataset, and all of these locations can be viewed in the Gantt chart if the user selects all 6 people in the adjacency matrix. However, if the user selects only the cells in the matrix, the filtering results in only 7 locations being shown in the Gantt chart, where meetings of at least two people occurred (Figure 11). We clearly see multiple group meetings at the "University", a group meeting at the "Thai Restaurant" and another at the "Mexican Restaurant", and another meeting between just two people at the "Indian Restaurant". The first day in Figure 11 shows four people meeting at the University, then 33 minutes later these four people join the "blue" person for lunch at the Thai Restaurant, then 45 minutes later the group meets the "yellow" person at the University. Then 4 days pass (shown folded), and on the 5th day another meeting occurs, etc.

6.3 Case Study 3: GeoLife data

We examined subsets of the GeoLife [53] dataset, containing over 180 people tracked over several months. Figure 12 shows meetings for a subset of 12 people that were loaded. We easily notice that 3 of these people have met an unusually high number of times. The matrix allows the user to easily filter down to any pair of these people. Examination of the "orange 3" and "pink 4" indicate that they may be close friends, as they are often together (Figure 13). (In the GeoLife data, these two people have IDs 003 and 004.)

However, examining the activity of "orange 3" and "blue 0" reveals that they seem to have *identical* activity patterns over large spans of time (Figure 14). (In the GeoLife data, these two have IDs 003 and 000.) After discovering this, we checked the raw data, and found that 129 of the original files for these individuals seemed identical. This points to an interesting anomaly worthy of further investigation. Were portions of the data accidentally duplicated?

7 Conclusion

We have presented MovementSlicer, a tool designed for understanding individual behaviors and meetings between small groups of people. Our taxonomy of visualizations (Figure 1), proposed categories of tasks (Section 3), and analysis of design options suggest that two kinds of Gantt charts are particularly suited for such

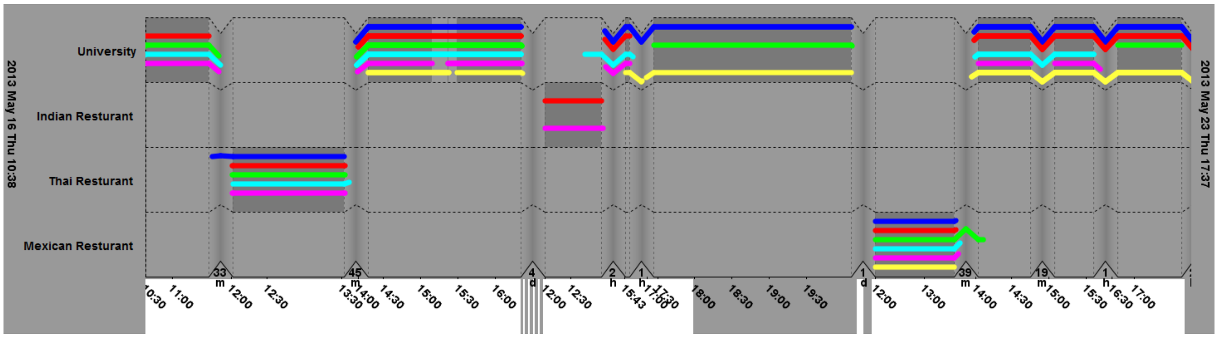

Figure 11: Case study 2: meetings between 6 members of our lab.

a tool, namely person-centric and location-centric, both of which show two categorical variables with respect to time. Gantt charts eliminate occlusion, making ordering of events and patterns such as repeated visits clearer, and also make it easy to show movements at very different spatial scales, e.g., between two buildings and between two cities in the same dataset.

Our prototype's scalability is improved using multiple strategies. To work better with more people or more moving objects, we support filtering via an adjacency matrix of meetings (this matrix having been automatically ordered, to make clusters of related people easier to see), and we also support excentric labels. To work better with long time spans, we support multi-focal temporal folding. To work better with many locations, we allow rows to be sorted by activity level. However, our prototype's user interface does not scale to hundreds of people or locations, nor thousands of meetings. For such cases, aggregation is necessary, with a Gantt chart as a possible option for subsets of data.

8 Future Directions

Gantt charts are best used on small numbers of moving objects. A more scalable system could combine multiple coordinated views, with detail-oriented views (such as Gantt charts) complementing overview-oriented views (such as the kind of aggregated transition diagrams in [5]). Evaluation with more datasets, and also with different users, would shed light on the most effective design.

There are also future avenues for improving coordination between views. We have started experimenting with displaying thumbnails of geographic map regions *within* the Gantt chart whenever the mouse cursor rolls over an event. (This is comparable to the use of embedded maps in [43].) We have also begun designing a kind of lens that could be passed over the geographic map, that would display the activity under the lens as a distance $\times$ time graph, i.e., like a small version of [13]'s graph, where distance could be measured with respect to the center of the map region under the lens. Such a lens could be useful for seeing when and how many times different people visit a given region of the map.

To increase flexibility, a kind of "generalized Gantt chart" could allow the user to map the horizontal and vertical axes to different quantities. We have already demonstrated the ability to nest person within location or location within person along the vertical axis. However, other systems [22, 31] allow arbitrary associations between data and axes. In the context of movement data, one useful mapping could be time along the horizontal, and *duration* of events along the vertical [35, 38], to make longer meetings or events stand out from others. Another would be the ability to map time-of-day to one axis, and date to another axis (as done with the "pixel based road speed views" in [49] and "two-dimensional time histograms" in [5]), or more generally, mapping (time modulo P) to one axis, and $\lfloor$ time / $P \rfloor$ to another, where P is some period (P could be 24 hours, or 7 days, etc.).

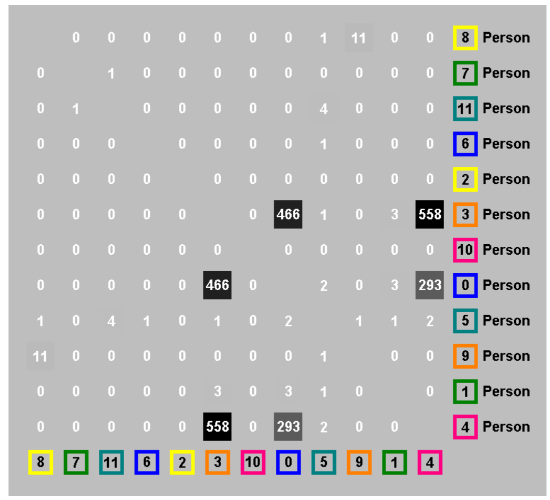

Figure 12: Case study 3: an adjacency matrix for several people from the GeoLife dataset.

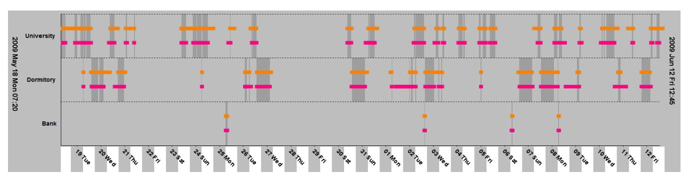

Figure 13: Case study 3: two people often together, at a university, at a dormitory, and at a bank.

Acknowledgements

Thanks to Renaud Blanch, Pourang Irani, and members of the HIFIV research group at ETS. This work was funded by NSERC.

References

[1] F. Amini, S. Rufiange, Z. Hossain, Q. Ventura, P. Irani, and M. J. McGuffin. The impact of interactivity on comprehending 2D and 3D visualizations of movement data. *IEEE TVCG*, 21(1):122–135, 2015.

[2] G. Andrienko, N. Andrienko, P. Bak, D. Keim, S. Kisilevich, and S. Wrobel. A conceptual framework and taxonomy of techniques for analyzing movement. *J. Visual Languages & Computing*, 22(3), 2011.

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

[4] N. Andrienko and G. Andrienko. Visual analytics of movement: An overview of methods, tools and procedures. *Information Visualization*, 12(1):3–24, 2013.

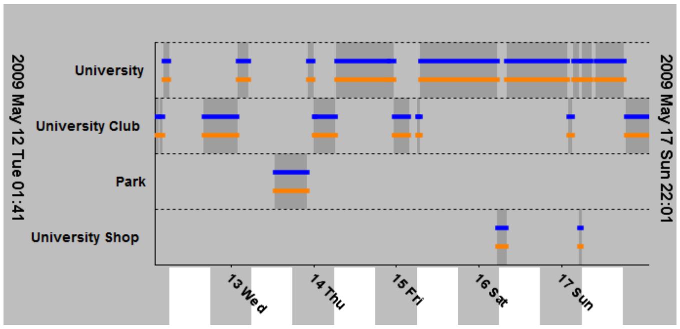

Figure 14: Case study 3: two people with identical data.

[5] N. Andrienko, G. Andrienko, G. Fuchs, and P. Jankowski. Scalable and privacy-respectful interactive discovery of place semantics from human mobility traces. *Information Visualization*, 2015.

[6] N. Andrienko, G. Andrienko, M. Wachowicz, and D. Orellana. Uncovering interactions between moving objects. In *Proc. International Conference GIScience*, pages 16–26, 2008.

[7] B. Bach, P. Dragicevic, D. Archambault, C. Hurter, and S. Carpendale. A review of temporal data visualizations based on space-time cube operations. In *Proc. EuroVis*, 2014. http://spacetimecubevis.com/.

[8] S. Bremm, T. von Landesberger, G. Andrienko, N. Andrienko, and T. Schreck. Interactive analysis of object group changes over time. In *International Workshop on Visual Analytics*, 2011.

[9] K. Buchin, M. Buchin, M. van Kreveld, M. Löffler, R. I. Silveira, C. Wenk, and L. Wiratma. Median trajectories. *Algorithmica*, 66(3):595–614, 2013.

[10] A. Buja, J. A. McDonald, J. Michalak, and W. Stuetzle. Interactive data visualization using focusing and linking. In *IEEE VIS*, 1991.

[11] A. Cockburn, A. Karlson, and B. B. Bederson. A review of overview+detail, zooming, and focus+context interfaces. *ACM Computing Surveys (CSUR)*, 41(1), 2008.

[12] C. Collins and S. Carpendale. VisLink: Revealing relationships amongst visualizations. *IEEE TVCG*, 13(6):1192–1199, 2007.

[13] T. Crnovrsanin, C. Muelder, C. Correa, and K.-L. Ma. Proximity-based visualization of movement trace data. In *IEEE VAST*, 2009.

[14] N. Elmqvist and J.-D. Fekete. Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE TVCG*, 16(3):439–454, 2010.

[15] N. Elmqvist, N. Henry, Y. Riche, and J.-D. Fekete. Mélange: space folding for multi-focus interaction. In *Proc. ACM CHI*, 2008.

[16] J.-D. Fekete and C. Plaisant. Excentric labeling: Dynamic neighborhood labeling for data visualization. In *Proc. ACM CHI*, 1999.

[17] H. L. Gantt. *Organizing for work*. Harcourt, Brace & Rowe, 1919.

[18] Gigantt. www.gigantt.com, 2014. http://youtu.be/XiD4ga2s13w.

[19] H. Guo, Z. Wang, B. Yu, H. Zhao, and X. Yuan. TripVista: Triple perspective visual trajectory analytics and its application on microscopic traffic data at a road intersection. In *Proc. IEEE PacificVis*, 2011.

[20] T. Hägerstrand. What about people in regional science? *Regional Science Association Papers*, 24(1):7–21, 1970.

[21] M. Hlawatsch, M. Burch, and D. Weiskopf. Visual adjacency lists for dynamic graphs. *IEEE TVCG*, 20(11):1590–1603, 2014.

[22] C. Hurter, B. Tissoires, and S. Conversy. FromDaDy: Spreading aircraft trajectories across views to support iterative queries. *IEEE TVCG*, 15(6):1017–1024, 2009.

[23] K. Isaacs, P.-T. Bremer, I. Jusufi, T. Gamblin, A. Bhatele, M. Schulz, and B. Hamann. Combing the communication hairball: Visualizing large-scale parallel execution traces using logical time. *TVCG*, 2014.

[24] J. Jo, J. Huh, J. Park, B. Kim, and J. Seo. LiveGantt: Interactively visualizing a large manufacturing schedule. *IEEE TVCG*, 20(12), 2014.

[25] T. Kapler and W. Wright. GeoTime information visualization. In *Proc. IEEE InfoVis*, 2004.

[26] M.-J. Kraak. The space-time cube revisited from a geovisualization perspective. In *Proc. International Cartographic Conference*, 2003.

[27] R. Krüger, D. Thom, M. Wörner, H. Bosch, and T. Ertl. TrajectoryLenses: A set-based filtering and exploration technique for long-term trajectory data. *CGF*, 32(3pt4):451–460, 2013.

[28] M.-P. Kwan. Gender, the home-work link, and space-time patterns of nonemployment activities. *Economic Geography*, 75(4), 1999.

[29] J. LeBlanc, M. O. Ward, and N. Wittels. Exploring N-dimensional databases. In *Proc. IEEE VIS*, pages 230–237, 1990.

[30] H. Liu, Y. Gao, L. Lu, S. Liu, H. Qu, and L. M. Ni. Visual analysis of route diversity. In *Proc. IEEE VAST*, pages 171–180, 2011.

[31] J. D. Mackinlay, P. Hanrahan, and C. Stolte. Show me: Automatic presentation for visual analysis. *IEEE TVCG*, 13(6):1137–1144, 2007.

[32] J. D. Mackinlay, G. G. Robertson, and S. K. Card. The perspective wall: Detail and context smoothly integrated. In *Proc. CHI*, 1991.

[33] E. Mäkinen and H. Siirtola. The barycenter heuristic and the reorderable matrix. *Informatica*, 29(3):357–363, 2005.

[34] T. Mihalisin, J. Timlin, and J. Schwegler. Visualization and analysis of multi-variate data: A technique for all fields. In *Proc. VIS*, 1991.

[35] C. Muelder, F. Gygi, and K.-L. Ma. Visual analysis of inter-process communication for large-scale parallel computing. *TVCG*, 2009.

[36] C. North and B. Shneiderman. Snap-together visualization: A user interface for coordinating visualizations via relational schemata. In *Proceedings of Advanced Visual Interfaces (AVI)*, pages 128–135, 2000.

[37] C. Plaisant, B. Milash, A. Rose, S. Widoff, and B. Shneiderman. LifeLines: visualizing personal histories. In *Proc. ACM CHI*, 1996.

[38] Y. Qiang, M. Delafontaine, M. Versichele, P. De Maeyer, and N. Nico Van de Weghe. Interactive analysis of time intervals in a two-dimensional space. *Information Visualization*, 11(4):255–272, 2012.

[39] J. C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *CMV*, pages 61–71, 2007.

[40] M. Steinberger, M. Waldner, M. Streit, A. Lex, and D. Schmalstieg. Context-preserving visual links. *IEEE TVCG*, 17(12), 2011.

[41] K. Sugiyama, S. Tagawa, and M. Toda. Methods for visual understanding of hierarchical system structures. *IEEE Trans. SMC*, 1981.

[42] Y. Tanahashi and K.-L. Ma. Design considerations for optimizing storyline visualizations. *IEEE TVCG*, 18(12):2679–2688, 2012.

[43] A. Thudt, D. Baur, and S. Carpendale. Visits: A spatiotemporal visualization of location histories. In *Proc. EuroVis, Short Papers*, 2013.

[44] M. Tory, S. Staub-French, D. Huang, Y.-L. Chang, C. Swindells, and R. Pottinger. Comparative visualization of construction schedules. *Automation in Construction*, 29:68–82, 2013.

[45] J. Trümper, J. Bohnet, and J. Döllner. Understanding complex multithreaded software systems by using trace visualization. In *Proc. SoftVis*, pages 133–142, 2010.

[46] Q. Ventura and M. J. McGuffin. Geo-Topo maps: hybrid visualization of movement data over building floor plans and maps. In *Proc. Graphics Interface (GI)*, pages 159–166, 2014.

[47] T. von Landesberger, S. Bremm, N. Andrienko, G. Andrienko, and M. Tekušová. Visual analytics methods for categoric spatio-temporal data. In *Proc. IEEE VAST*, pages 183–192, 2012.

[48] K. Vrotsou, J. Johansson, and M. Cooper. Activitree: interactive visual exploration of sequences in event-based data using graph similarity. *IEEE TVCG*, 15(6):945–952, 2009.

[49] Z. Wang, M. Lu, X. Yuan, J. Zhang, and H. van de Wetering. Visual traffic jam analysis based on trajectory data. *IEEE TVCG*, 2013.

[50] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *AVI*, 2000.

[51] W. Zeng, C.-W. Fu, S. M. Arisona, and H. Qu. Visualizing interchange patterns in massive movement data. *CGF*, 32, 2013.

[52] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang. Collaborative location and activity recommendations with GPS history data. In *WWW*, 2010.

[53] Y. Zheng, X. Xie, and W.-Y. Ma. GeoLife: A collaborative social networking service among user, location and trajectory. *IEEE Data Engineering Bulletin*, 33(2):32–39, 2010.