International Journal of
Geo-Information

MDPI

Review

Diverse Visualization Techniques and Methods of Moving-Object-Trajectory Data: A Review

Jing He [1], Haonan Chen [1,*], Yijin Chen [1,*], Xinming Tang [2] and Yebin Zou [1]

[1] College of Geoscience and Surveying Engineering, China University of Mining and Technology (Beijing), Beijing 100083, China; hejing@student.cumtb.edu.cn (J.H.); tbp1600202041@student.cumtb.edu.cn (Y.Z.)

[2] Satellite Surveying and Mapping Application Center, National Administration of Surveying, Mapping and Geoinformation of China, Beijing 100048, China; txm@sasmac.cn

[*] Correspondence: harman@student.cumtb.edu.cn (H.C.); y.j.chen@cumtb.edu.cn (Y.C.); Tel.: +86-130-5188-8722 (H.C.)

Received: 18 November 2018; Accepted: 23 January 2019; Published: 29 January 2019

Abstract: Trajectory big data have significant applications in many areas, such as traffic management, urban planning and military reconnaissance. Traditional visualization methods, which are represented by contour maps, shading maps and hypsometric maps, are mainly based on the spatiotemporal information of trajectories, which can macroscopically study the spatiotemporal conditions of the entire trajectory set and microscopically analyze the individual movement of each trajectory; such methods are widely used in screen display and flat mapping. With the improvement of trajectory data quality, these data can generally describe information in the spatial and temporal dimensions and involve many other attributes (e.g., speed, orientation, and elevation) with large data amounts and high dimensions. Additionally, these data have relatively complicated internal relationships and regularities, whose analysis could cause many troubles; the traditional approaches can no longer fully meet the requirements of visualizing trajectory data and mining hidden information. Therefore, diverse visualization methods that present the value of massive trajectory information are currently a hot research topic. This paper summarizes the research status of trajectory data-visualization techniques in recent years and extracts common contemporary trajectory data-visualization methods to comprehensively cognize and understand the fundamental characteristics and diverse achievements of trajectory-data visualization.

Keywords: trajectory set; spatiotemporal information; attribute information; hidden information; techniques and methods; research status

1. Introduction

Current trajectory data are ubiquitous, involving abundant datasets with multiple dimensions and many variables; the relevant research has facilitated the comprehension of dynamically evolving behaviors and the prediction of future moving trends. Trajectory-data visualization involves modeling through computer graphics, computer vision, user interface and other techniques to visually represent trajectory data and provide effective interaction to support users' data-exploration behavior [1]. After years of research, the field has achieved considerable results worldwide, establishing appropriate visual representations for the dimensional information in trajectory data. However, the diversity of trajectory-visualization techniques could cause problems for researchers and designers in terms of clearly comprehending what has been accomplished, how to compare and choose visualization methods, and how much potential of space–time data visualization remains to be explored. Therefore, this work deserves to be composed and organized.

First, we should fully understand the conceptual distinction between "multidimensional" and "multivariate". Generally, "multidimensional" is used to describe the dimensionality of independent variables in physical space and concentrates on expressing spatial and temporal concepts, while "multivariate" is used for attribute-count description that indicates the attribute amount involved in the data. In fact, trajectory data are typically enormous, high-dimensional, spatiotemporal vector attribute data. We are interested not only in comprehending dimensional information for spatiotemporal-element depictions, but also in the visualization research with attribute-feature expression. Starting in Section 2, we present relevant visualization techniques and methods that function with multisource trajectory data. The chosen examples are for illustration and the list is not exhaustive—we only focus on small but representative choices.

2. Universal Multivariate Visualization

The trajectory visualization reflects the changing patterns of time-varying geolocations and corresponding attribute information, including basic-dimensional information and the attached thematic variables of space–time objects. However, these variables usually do not exist independently but are combined with each other, and the trajectory data of different sources, scales, or types are plotted in a hybrid manner. We separately list visualization techniques that integrate multiple attributes. As a powerful and portable carrier that encodes abundant information in methods of visual channels and image fusion, these techniques are rather common in directly or indirectly visualizing multiattribute information of trajectory data.

2.1. Icons

Icon method represents multiattribute trajectory data by using icons with multiple, easily identifiable features, each of which represents its corresponding univariate information. Specifically, icons map data attributes into different visual channels (icons, textures, colors and transparency, outlines, shapes, metaphors, etc.) and then combine them with other visualization stages in a reasonable manner, such as mixed rendering in the plotting stage, image fusion in the image-processing stage, etc. Figure 1 is the "Whisper" design by Cao et al. [2], which is based on a metaphor of a sunflower, tracing the paths from the source tweets to different groups of users who retweeted them.

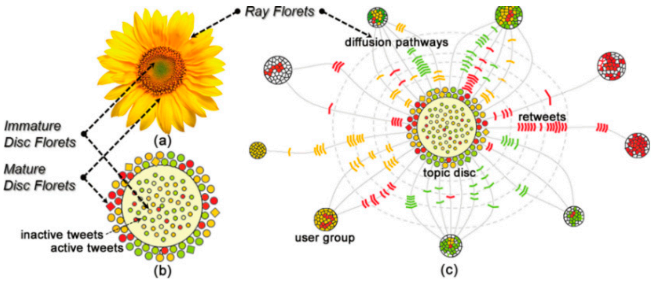


Figure 1. Visual metaphors of Whisper [2].

1. Geometric Icons

A feasible approach to visualize trajectories with attributes is to represent the data items as geometric icons and map data attributes to visual properties, which is performed by selecting a mapping configuration for each data item that describes the parameterization of geometry and mapping and then applying this configuration to its attributes. As proposed by Buschmann et al. [3], the technique of real-time animation that visualizes a large amount of trajectory data supports the general mapping of trajectory attributes to geometric representations and appearances. The central function of this visualization method consists of the efficient rendering of 3D movement trajectories and their attributes. This method maps geometric icons (lines, bands, tubes or spheres) to visual properties such as color, linewidth, or the diameter of tubes or spheres. These geometric representations are also supplemented by texture mapping [4] and animation. Figure 2 shows the fundamental icon types provided by this visualization technique and the interactive application of this approach to real-time animated 3D trajectory visualization.

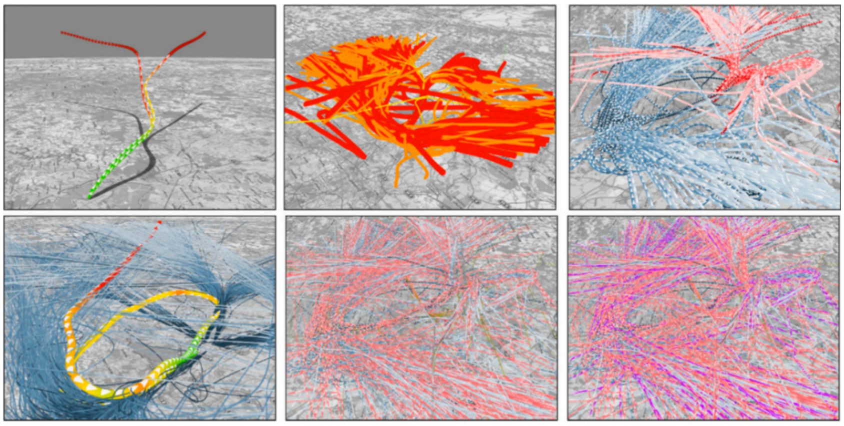


Figure 2. Interactive visualizations based on geometric icons [3].

2. Radial Icons

To intuitively reflect spatial distribution and multidimensional attributes in path visualizations, Sheng [5] designed a wheel-metaphor icon based on a radial layout in an area attribute view, which corresponds to the candidate area and maps the multivariable attributes simultaneously. Meanwhile, mapping key areas to the icons of the attribute-visualization interface often requires the position updating of icons to alleviate visual cluttering caused by the distance reduction between the icons, or sequentially mapping grid areas according to the increasing or decreasing order of abscissas of the area centers, solving the node-overlapping problem. Figure 3 shows the mapping effect of certain key areas to icons in a certain period.

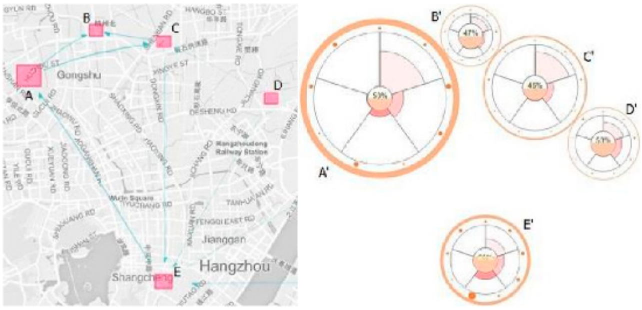
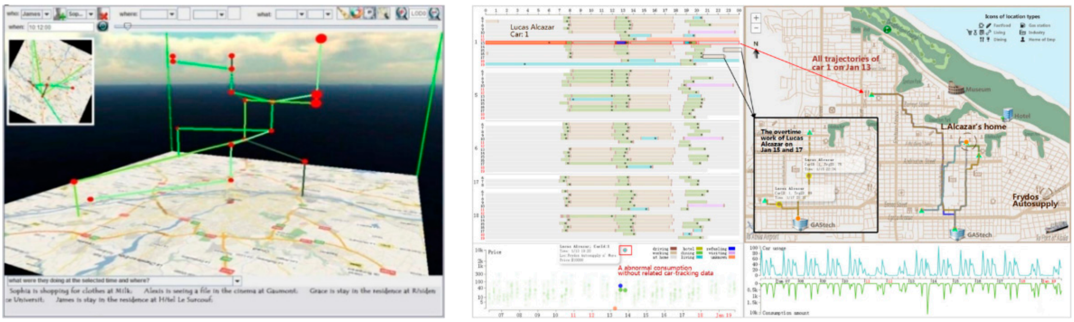


Figure 3. Example of radial icon visualization [5].

2.2. *Semantics*

1. Trajectory Semantics Study

Semantics refers to the available context information regarding a moving object in addition to its location data, which is included in the geometric properties of the spatiotemporal stream and the geospatial area where the object moves. Spaccapietra et al. [6] proposed the first model to deal with trajectory data from a semantic perspective. Converting a sampled trajectory into a semantic one (sequence of stops and movements) is not an easy task. For example, Zhong et al. [7] extracted semantic trajectory patterns at different spatial and temporal granularities, reflecting the detection of spatiotemporal events (see Figure 4a). Crime map [8,9] is a typical application of introducing semantic analytics into the field of digital forensics and criminal investigation. Liao et al. [10] designed a comprehensive analytic system for trajectory data and transaction data, which enables multiple perspectives in a semantically enhanced manner, depicting individual life routines and detecting suspects' behaviors. As shown in Figure 4b, Lucas Alcazar was listed as a criminal suspect according to the charge record, personal activity and movement path.


who: James
where:
what:
when:
Lucas Alcazar
Car: 1
All trajectories of car 1 on Jan 13
Museum
Hotel
Alcazar's home
The overtime work of Lucas Alcazar on Jan 15 and 17
Frydos Autosupply
GAStech
A abnormal consumption without related car-tracking data
Price
Car usage
Consumption amount

what were they doing at the selected time and where?

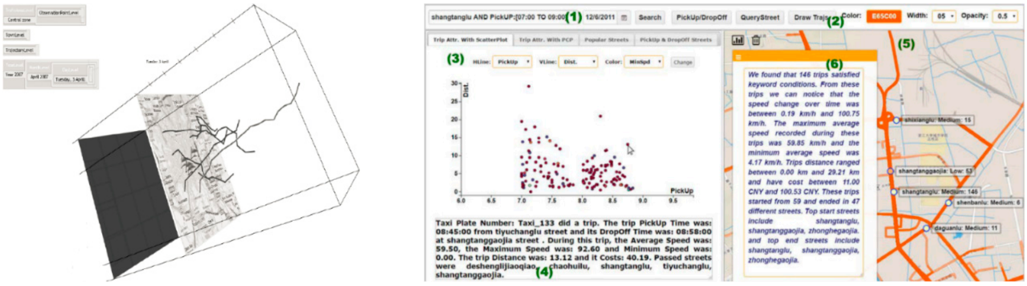
Sophia is shopping for clothes at Milk. Alexis is seeing a file in the cinema at Gaumont. Grace is stay in the residence at R/siden ce Universit. James is stay in the residence at H/tel Le Surcouf.

(**a**) (**b**)

Figure 4. Examples of trajectory semantic visualization: (**a**) trajectory semantic-query system [7]; and (**b**) abnormal transactions and overtime work patterns of a criminal suspect [10].

2. Trajectory Semantic-Pattern Exploration

For each application domain, the trajectory has a set of specific features and requires an approach that provides the correct visual cues while integrating with different semantic data. In response to this work, Le et al. [11] proposed the first demonstrable interactive semantic visualization system, "SemVis", which infuses visualization with latent semantics to build an application ontology. Each trajectory in the ontology is considered to have the following semantic attributes: (a) being located in one or several neighborhoods; (b) occurring during a specific day; and (c) possibly passing through some observation points in various traffic areas. Figure 5a presents a user-defined trajectory cube that associates the trajectory with recognized meaningful semantic dimensions. Furthermore, novel interfaces are required to transform abundant trajectory data into different spatial and temporal granularities. SemanticTraj, which was designed by Al-Dohuki et al. [12], enables an efficient visual exploration of taxi trajectories (see Figure 5b).



Taxi Plate Number: Taxi_133 did a trip. The trip PickUp Time was: 08:45:00 from tiyuchanglu street and its DropOff Time was: 08:58:00 at shangtanggaojia street . During this trip, the Average Speed was: 59.50, the Maximum Speed was: 92.60 and Minimum Speed was: 0.00. The trip Distance was: 13.12 and it Costs: 40.19. Passed streets were deshenglijiaoqiao, chaohuilu, shangtanglu, tiyuchanglu, shangtanggaojia.

We found that 146 trips satisfied keyword conditions. From these trips we can notice that the speed change over time was between 0.19 km/h and 100.75 km/h. The maximum average speed recorded during these trips was 59.85 km/h and the minimum average speed was 4.17 km/h. Trips distance ranged between 0.00 km and 29.21 km and have cost between 11.00 CNY and 100.53 CNY. These trips started from 59 and ended in 47 different streets. Top start streets include shangtanglu, shangtanggaojia, zhonghegaojia. and top end streets include shangtanglu, shangtanggaojia, zhonghegaojia.

(**a**) (**b**)

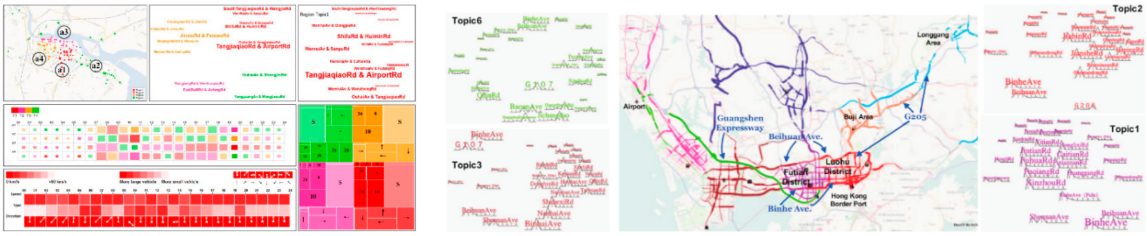
Figure 5. Examples of analyzing trajectory semantic patterns: (**a**) encoding downtown pedestrian trajectories by semantic types [11]; and (**b**) visually exploring taxi trajectories via SemanticTraj [12–14].

2.3. Word Clouds

1. Subject Content

As the latest variant of semantics, word clouds, a fundamental visual metaphor, are a feature-based text visualization, where a feature indicates a non-overlapping text chunk (e.g., keywords or phrases) or a grammatical structure (e.g., infinitives or clauses) within a document. The typical word-cloud graph highlights essential keywords in massive texts, indicating moving-object attributes with font size or overall layout. Because of the popularization of word clouds, many recent studies have examined this subject, whose topics ranged from usability issues [15] through dynamic text visualizations [16] to the ranking evaluation of Twitter-trending words that addresses anomalous event detected from the Twitter stream [17–21].

In recent years, the development of smart traffic management systems has built denser checkpoints on urban roads. Figure 6a shows trajectory information recorded by checkpoints for all vehicles per unit of time. Chu et al. [22] designed a novel VATT (Visual Analytics of Taxi Topics) system that transforms geographic coordinates in massive trajectory data of urban traffic networks into taxi topics with street clouds. Figure 6b shows the taxi topics generated from massive taxi-trajectory sets, which were ranked and labeled according to their importance from Topic 1 (high importance) to Topic 8 (low importance).



(**a**) (**b**)

Figure 6. Word-cloud views of trajectories: (**a**) traffic areas indicated by word clouds [23]; and (**b**) taxi-topic visualization [22].

2. Multilayered Geo-Locational Word Clouds

Generally, visualizing global events via numerous parallel small views can result in incomplete visualization in the limited screen space. In this regard, Itoh et al. [24] proposed a method for visualizing spatiotemporal trajectory events by a multilayered geo-locational word-cloud representation from a geo-located microblog stream. This stacked multilayer design visualizes multiple layers in 3D space with a 2.5D representation, detecting where the trajectory events occurred and placing the global events on multiple individual layers. Figure 7 illustrates the huge global event of the Tokyo Marathon, alongside related events and their temporal evolution. The locations of trajectory events in each snapshot were strongly affected by the positions of the runners.

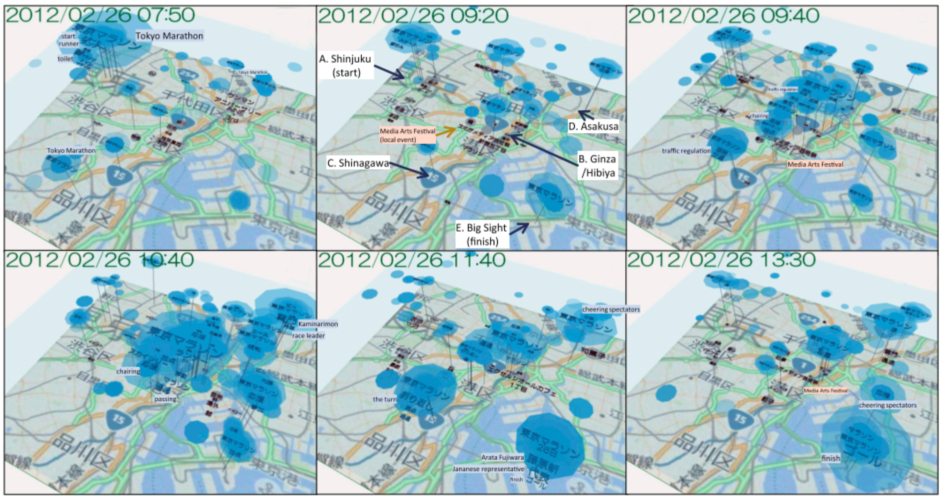


Figure 7. Multilayered spatial word-cloud visualization of the Tokyo Marathon on 26 February 2012 [24].

3. Visualization Targeting Low-Dimensional Data

Without further explanation, we consider trajectory data below four dimensions as the major research target of low-dimensional trajectory data. Diverse trajectory data visualizations exist, which principally involve spatial dimensions, sometimes temporal dimension. However, the diversity of variable-based methods, such as visual encoding of the icon method, is attached to dimension-based visualizations. We illustrate attributes and accompanied complex features such as multiscale and dynamic in low-dimensional visualization techniques of trajectory data, which include at least one variable.

3.1. Space–Time Cube (STC)

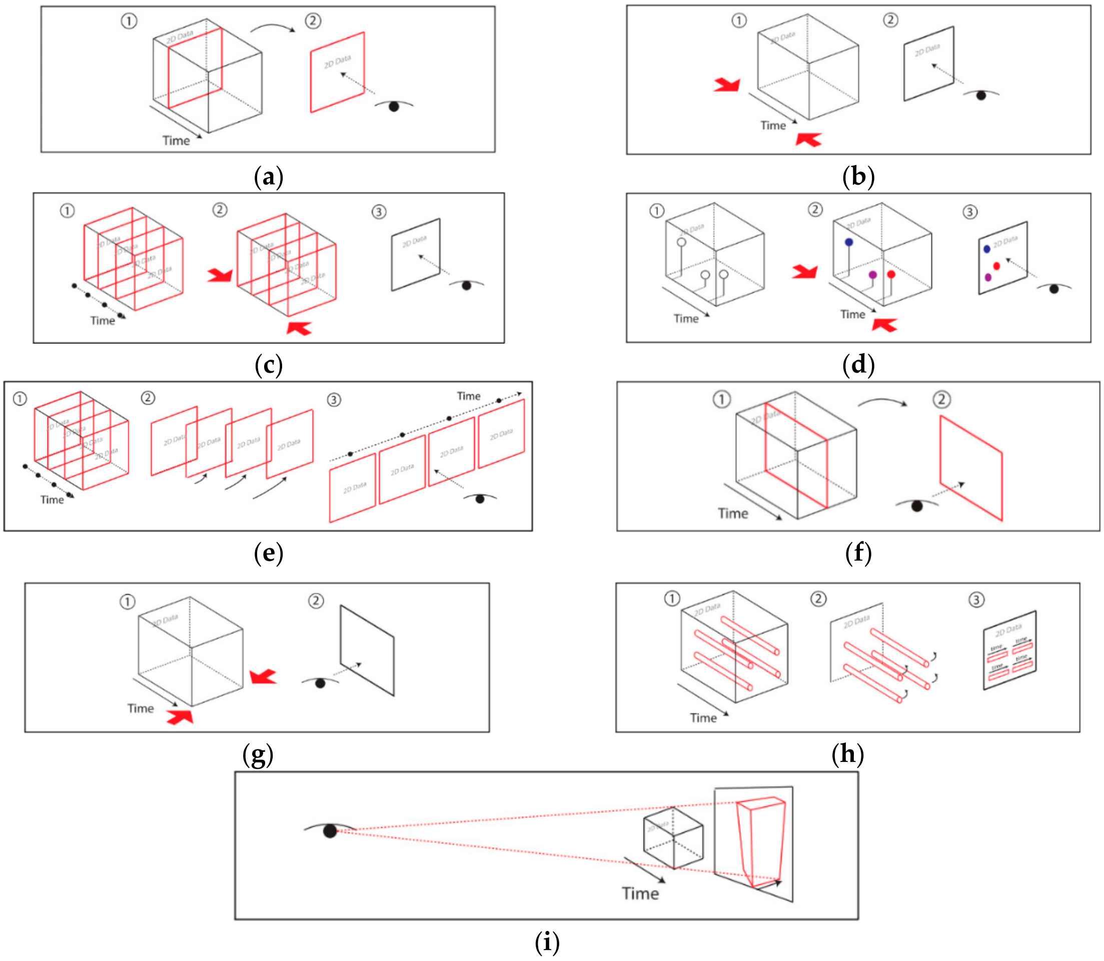
1. Model

Space–time cube refers to a three-dimensional Euclidean space that consists of two geospatial dimensions and a discrete or continuous temporal dimension, enabling the expansion of two-dimensional space and explicit memory of time information [25]. As a visualization of three-dimensional spatiotemporal trajectories, this method has attracted the attention of many scholars. It is both an event-state-oriented and event-process-oriented spatiotemporal data model. Information such as the time and speed of the moving objects can be intuitively visualized through the three-dimensional space–time path.

2. Interactions

A space–time cube display typically supports static and dynamic interactions for the adjustment of viewpoint, which facilitate the perception of spatiotemporal patterns [26].

- Static Visualization: Static interactions of generalized STCs (basic and composite) are used to depict various operations of spatiotemporal data [27], including time cutting (extracting a particular temporal snapshot from the cube), time flattening (collapsing the cube along its time axis by merging all time slices into a single view), discrete time flattening (similar to time flattening but selecting target time slices before flattening instead of merging them all), colored time flattening (similar to time flattening but the time slices are colored before combination), time juxtaposing (cutting multiple time slices and placing these slices side by side or on a grid), space cutting (extracting a planar slice that is perpendicular to the data plane), space flattening (similar to space cutting but consists in flattening the cube in a particular direction on the data plane instead of extracting a space cut), repeated drilling (extracting drilling cores at several locations on the visualization plane and rotating these cores in-place) and 3D rendering (projecting 3D objects onto a 2D plane). Figure 8 presents schematics that focus on static-visualization operations.
- Dynamic Visualization: Animation is the process of applying different operations on a space–time cube over time, or similarly, varying the operation parameters over time. The most common forms of animation involve changing the position of the cutting operation over time (i.e., animated time cutting). Spatial padding can be performed accordingly before the operation, producing smooth animated transitions. Animated time cutting can also be combined with other STC operations, such as time flattening. Although many animation techniques can be described as animated time cutting and its variants on a static space–time cube, other animation operations also exist. For example, Bach et al. [28] proposed animated 3D rendering, which interprets the transition between two space–time cube operations by smoothly rotating the space–time cube representation.

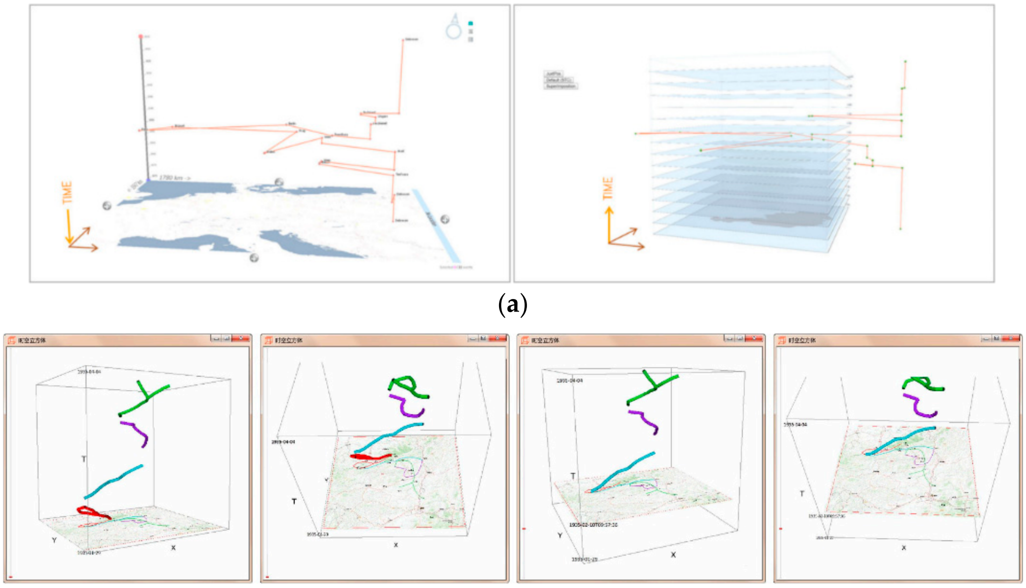


(**a**) (**b**) (**c**) (**d**) (**e**) (**f**) (**g**) (**h**) (**i**)

Figure 8. Static visualizations as space–time cube operations [27]: (**a**) time cutting; (**b**) time flattening; (**c**) discrete time flattening; (**d**) colored time flattening; (**e**) time juxtaposing; (**f**) space cutting; (**g**) space flattening; (**h**) repeated drilling; and (**i**) 3D rendering (image courtesy of Bach et al.).

3. System Exploration

Many visualization systems apply space–time cubes, including Geo-Time [29], PolyCube [30], Tardis [31], VISUAL-TimePAcTS [32], Cubix [28], Wakame [33], etc. For example, Mayr et al. [30] extracted the recording locations and corresponding time stamps of biographical narratives, which were then injected into spatiotemporal visualization systems for further exploration (see Figure 9a). The correct choice of space–time cube operations involves a trade-off between analytically related space–time characteristics, such as the density or sparsity of a cube. The space–time path (STP) refers to a three-dimensional space–time path that combines temporal information and supports specific applications of each interactive operation. Figure 9b shows a space–time cube that narrates the "Crossing the Chishui River Four Times" event, a famous battle in the Red Army Long March in 1935. A time-slice technique was applied to visualize two-dimensional temporal snapshots that correspond to different temporal points in the three-dimensional scene, expressing the event status over time.



(**a**)

(**b**)

Figure 9. Applications of space–time cube interactions: (**a**) the Red Army's Crossing the Chishui River Four Times [34]; and (**b**) life path of the Austro-Hungarian opera singer and director, Joszef Szabo [30].

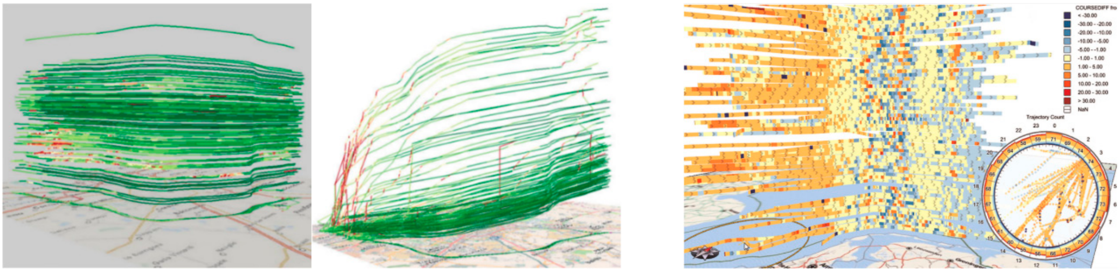
3.2. Stacking

1. Stacking Extension

A stacking graph denotes a piece of data as a band (or layer) and then sequentially stacks the resulting bands in a common baseline manner to represent a plurality of time-varying data. The amplitude of each band at a particular point in time represents the value of the corresponding time-varying data. Some researchers have focused on improving visual perception, such as ThemeRiver by Havre et al. [35], where layers indicate symmetrical stacking with smooth curves. In addition to improving aesthetics and legibility, researchers have studied how to improve the effectiveness of stacking graphics. For example, Wu et al. [36] designed PieceStack, an interactive system that focuses on supporting people in understanding the configuration of stacking graphs. Besides, enriching visual encoding and combining various interactions can provide new features for stacking graphs to process trajectory-set data. In other words, the fundamentals of applying stacking to trajectory-set data are merging the interrelated attribute information of moving objects and displaying them for each sampling point of all movement trajectories in a three-dimensional visualization effect to facilitate comparisons and statistics.

2. Trajectory Wall

Many methods are applicable to the basic tasks of trajectory visualization, but the distribution and dynamics of attribute values in space and time are still unclear, especially when the analyst is interested in not only a single trajectory but also the entire trajectory set. Inspired by Tuan et al.'s [37] stacking of graphic elements, Tominski et al. [38,39] devised a solution that visualizes attribute values along stacked trajectories. As shown in Figure 10a, this approach stacks trajectory bands along the z-axis perpendicular to the map by a temporal order, while providing an exclusive layer on the z-axis for each trajectory. Therefore, this method avoids the possible intersection of trajectory bands (the right graph), which would cause problems when tracking attributes along the trajectories. To illustrate the utility of trajectory stacking, Andrienko et al. [40] explored a trajectory set with similar shapes in a 3D view, where the trajectory bands were divided into segments that were colored according to their attribute values (see Figure 10b).



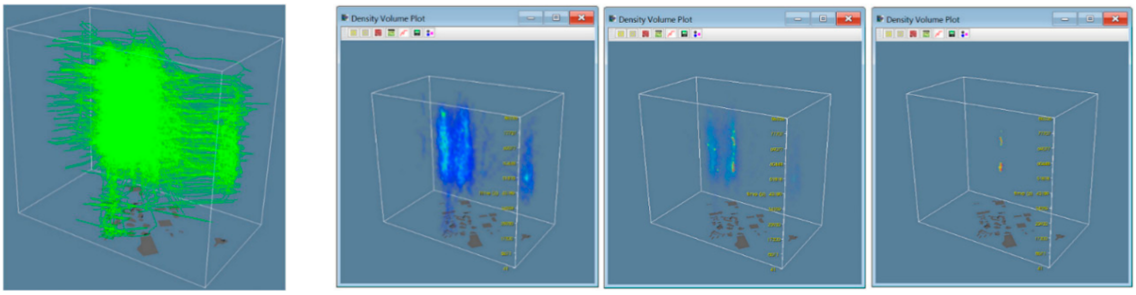
(**a**) (**b**)

Figure 10. Examples of trajectory-wall visualization: (**a**) comparison between trajectory wall and space–time cube [38,39]; and (**b**) vessel movements in the North Sea [40].

3.3. Density Map

1. Why Choose a Density Map?

As the availability of larger trajectory datasets grows, the three-dimensional STPs in space–time cubes inevitably suffer overlapping and cluttering, both of which obscure the visual recognition of spatiotemporal patterns. The spatiotemporal clustering of spatial-trajectory datasets that focus on detecting people's daily activity information can help explore the moments and locations of interactions. Figure 11a is a snapshot of numerous student tracks in a traditional STC visualized by Du et al. [41] that indicates the spreading risk of an epidemic disease, which demonstrates how the visual cluttering problem obscures the detection of any cluster.



(**a**) (**b**)

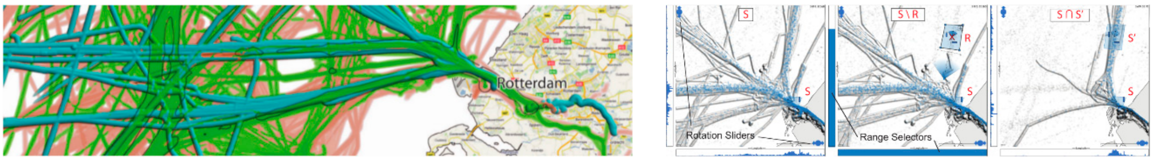
Figure 11. Applications of detecting epidemic-disease risk with: (**a**) space–time cube; and (**b**) density map [41].

Density maps can smoothly visualize such data instead. The authors resampled trajectory data with the same time interval and replotted them using a density map. Figure 11b shows how the density heatmap changes when the opacity of more voxels is set to zero. The rightmost map retains only high-density voxels to denote potential spatiotemporal clustering, unearthing the highest risk of flu transmission.

2. Application Case

- Establishing a Density Map

Density map is a high-level overview of traffic flow, which is widely used in the field of maritime transportation. For example, Scheepens et al. [42] composited multiple density maps to analyze the accident risk of marine traffic (see Figure 12a). Further, Scheepens et al. [43] applied an animated particle system to analyze the trajectories of vessels near the Rotterdam harbor in a density map form: the particle velocities were all initialized to a constant; each trajectory was rendered as an individual particle that moving along the original trajectory; and then all the trajectories were rendered together to generate a density map and mine the behavioral-pattern information covered in the traffic flows (see Figure 12b).



(**a**) (**b**)

Figure 12. Density-map visualizations of traffic trajectory data (**a**) composite density map of vessel-accident risks [42]; and (**b**) density map combined with a particle system [43].

- Trajectory 3D Space–Time Density

As introduced by Demšar et al. [44], the 3D space–time density of a trajectory is a generalization of the standard 2D kernel density around the 2D point data into the 3D density around the 3D polyline data (i.e., trajectories). Its fundamentals are first computing the density for each trajectory, which represents the influence of that particular trajectory on its nearest neighborhood in both space and time, and then adding these densities to obtain the total density for the entire dataset of trajectories. Demšar et al. also applied this approach to the analysis of vessel-movement trajectories in the Gulf of Finland. Their result (Figure 13) depicts the traffic characteristics of tanker vessels from four different perspectives.

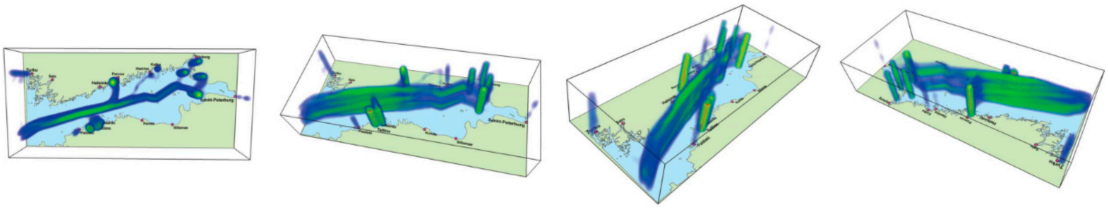


Figure 13. Space–time density of tankers displayed from different perspectives [44].

- Solving Over-Plotting

Interpreting the diffusion direction of mobile crowds can help urban designers optimize routes and traffic flows. Therefore, Li et al., from Hong Kong Polytechnic University, Hong Kong, China, proposed a StreamMap model [45] to smoothly blend high-density streaming points and create a visual flow that emphasizes the density-pattern distributions. To solve the over-plotting problem, a superpoint-based estimation method named super kernel density estimation (SKDE) is proposed to derive accurate density maps from the period of streaming point data by using adaptive kernel selection with a fast point-clustering method. Figure 14 depicts smooth morphing results of visualizing the pedestrian flow on 25 July 2015 at East Nanjing Subway Station via improved StreamMap model to detect flow trends.

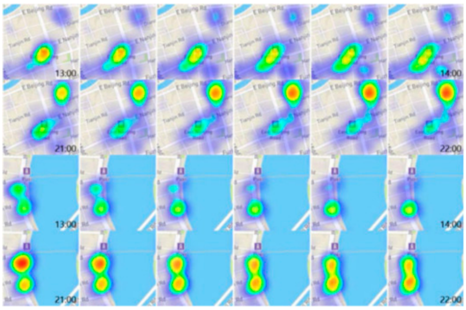


Figure 14. Morphing results of pedestrian flow visualization between two time steps [45].

3.4. Heatmap

1. How to Display?

Traditional point-data visualization directly marks the point objects based on their coordinate projection. However, this direct approach causes covering and overlapping as the data amount grows, and the screen utilization is inappropriate with an uneven data distribution. Heatmap (i.e., thermal map) aggregates discrete points into a continuous form according to the spatial location of the data and reassigns the heat values for each location to render geographic attribute values into colors or transparency [46], hence a qualitative analysis visualization method that displays statistical data instead of each point's data. The heatmap in Figure 15 depicts the density distribution of traffic flow by indicating the flow attributes with color intensities.

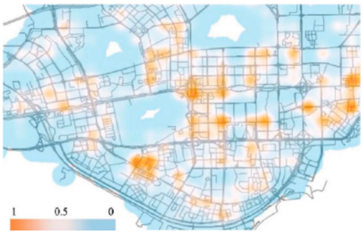


Figure 15. Hotspot detection with the heatmap technique [47].

2. Plotting Methods

From the perspective of heatmap generation, three main plotting methods exist: influence superimposition, density grids [48], and sliding windows [49]. The influence-superimposition approach is simple and widely adopted. The basic idea is to map the geographic objects onto the geographical space as hotspots, each having a certain influence on its neighborhood (determined by the kernel function), and then sum the influence of each hotspot and convert the added results into colors to render, yielding a heatmap. Embedding geolocation information into datasets in social media benefits analyses on location-based public behaviors. In terms of spatiotemporal analytics, Chae et al. [50] used a geospatial heatmap to illustrate the spatial distribution and trend approximation of public movement flow in the disaster event of Hurricane Sandy in Manhattan, New York in 2012. Figure 16 shows their user-based spatial tweet distribution in Manhattan; such standard heatmap visualization enables analysts to explore the spatial patterns of Twitter users at any given time.

However, general heatmap plotting exhibits some shortcomings, such as poor adaptability, monotonous accumulation mode of point influence, and insufficient consideration of geospatial characteristics. Based on the above-mentioned defects, Wang et al. [51] proposed a plotting method of reverse-rendering heatmaps that enables a more robust and comprehensive heatmap display. The authors applied this approach to passenger trajectory data for public transportation in Beijing during rush hours and analyzed the urban passenger-flow distribution (see Figure 17).

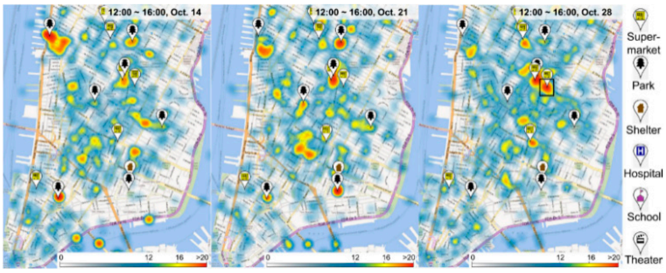


Figure 16. Heatmap views of spatial tweet distribution during a hurricane disaster [50].

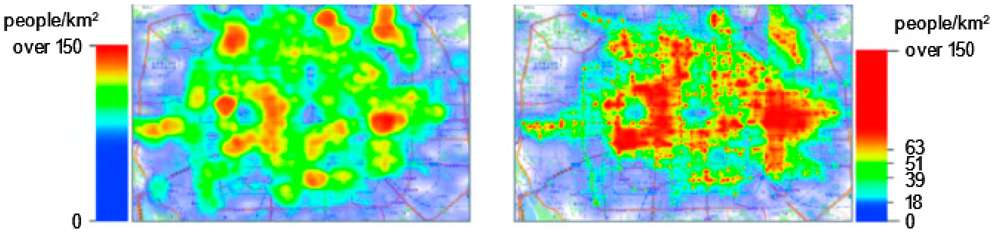
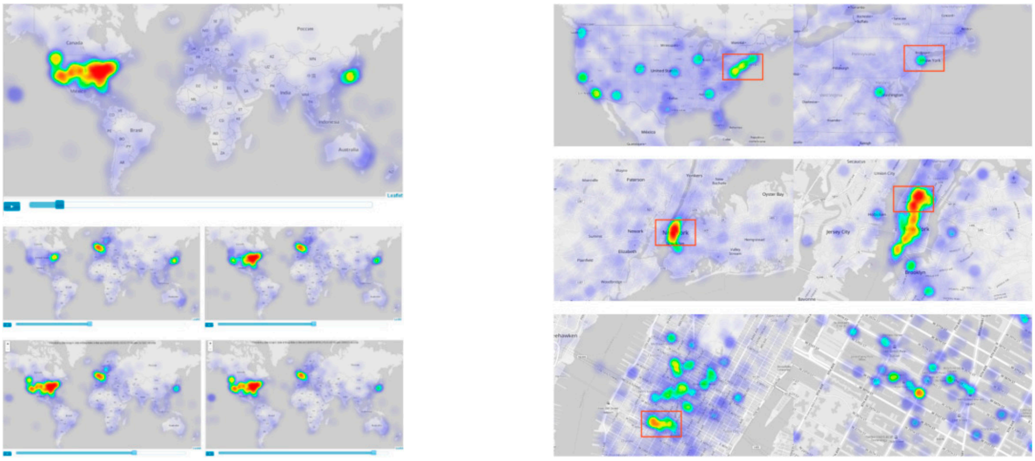


Figure 17. Comparison between general heatmaps (**left**) and reverse heatmaps (**right**) of Beijing urban area [51].

3. Interactive Visualization

Time-oriented data are typically defined as streaming data. In particular, streaming point data are a type of streaming data that store geographical positions, which also involves a portion of the trajectory data. Li et al. [52] proposed a perceptually optimized method to dynamically visualize high-density streaming points and introduced two extended interactions for the visualization of large point sets—stream player and hierarchical interaction. A stream player is an interactive environment for users to easily browse the results by playing a stream. Each calculated sequence is stored as an image, whose interpolation is dynamically calculated according to a linear function. Therefore, switching from one sequence to another is efficient (see Figure 18a). Zooming is an essential feature in interactive visualization. Users cannot browse all the details at the same time because the pipeline aims for large point sets. Hierarchical interaction was therefore developed, that is, sampling the dataset according to the current scale level and constructing a quadtree structure for the high-speed query of datasets in the selected area. Figure 18b shows six perceptual heatmap results at different levels of detail.



(**a**) (**b**)

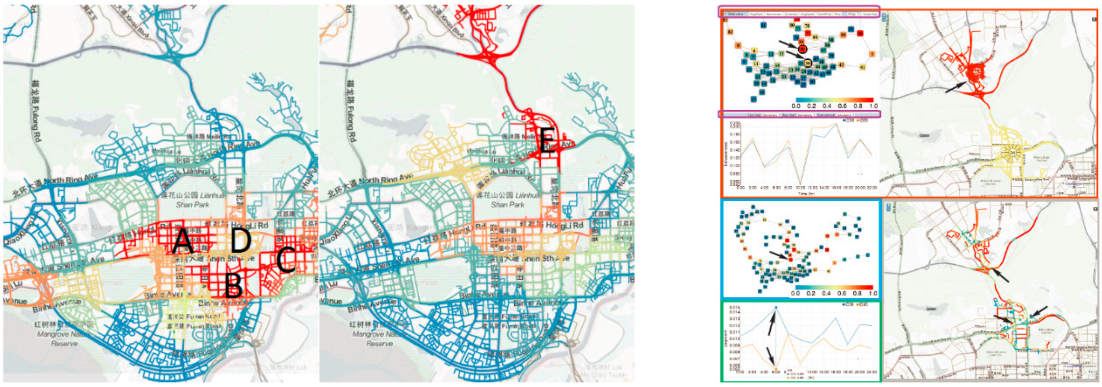
Figure 18. Example of interactive heatmap visualization using: (**a**) stream player; and (**b**) hierarchical interactive visualization [52].

3.5. Meshing

Meshing methods are a spatial-visualization technique that displays statistical data in a mesh pattern. When applied to trajectory set data, these methods refer to representing trajectory sampling data on a map according to their homogeneity in a certain mesh cell. If regarded as homogeneous, these data are directly assigned attributes and then visualized; if not, the set cells are subdivided into smaller mesh cells according to certain rules or requirements, and the data are distributed to each cell for visualization processing via comprehensively weighing its spatial distribution features. As a persuasive example and a novel mesh-based visualization method, TrajGraph [53], proposed by Huang et al. from Kent State University, has a good visualization effect when handling the scale and complexity of trajectory data and supporting domain users while interactively searching and comparing time-varying traffic information of urban roads with real traffic information.

1. TrajGraph

TrajGraph is aggregated by graph partitioning to represent region-level information, which calculates two indicators of “pagerank” and “betweenness” and merges real traffic information carried by taxi-trajectory data to reflect urban network structures. Figure 19a provides two examples of how the centralities reveal the transportation patterns of Shenzhen. Furthermore, TrajGraph effectively facilitates the interactive visual reasoning of real traffic conditions. As shown in Figure 19b, analysts may filter display areas by scoping city-wide graphs and zooming in graph views, acquiring contrary or similar temporal variations in the betweenness centrality of the day and further displaying the pagerank values of their different timelines to determine where urban bottleneck areas are located. If this pattern recurs for several days, then a solution should be sought in terms of urban planning or traffic control.

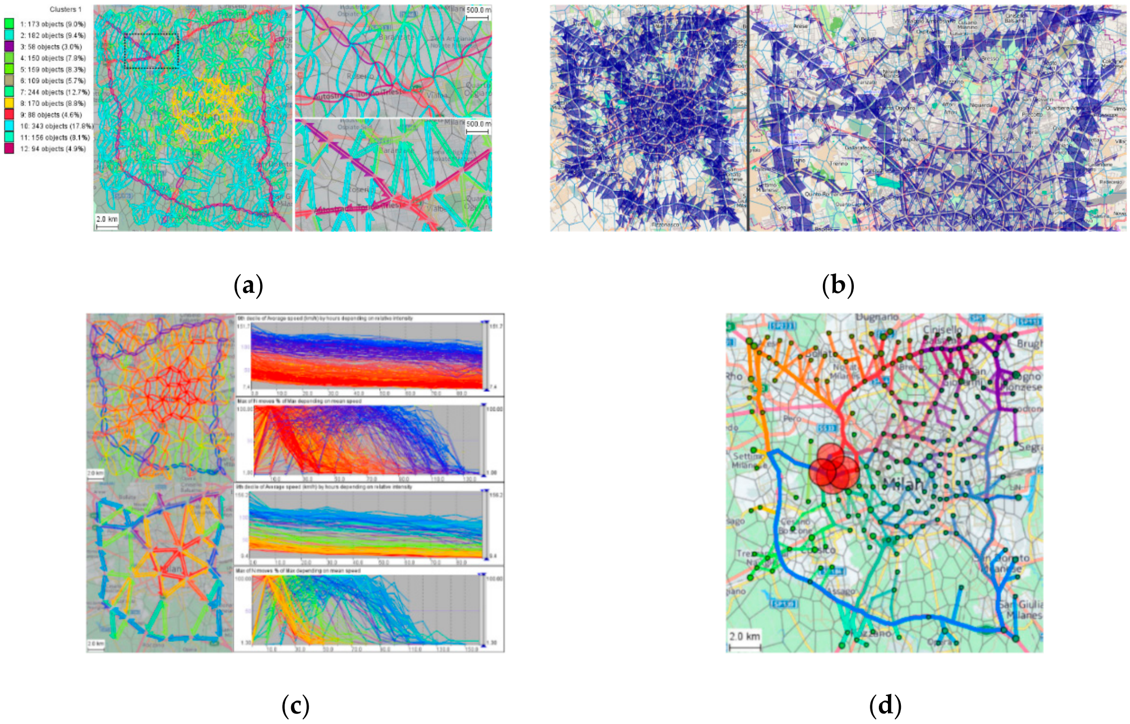


(a) (b)

Figure 19. TrajGraph visualization [53]: (**a**) filtering preferred hub regions; and (**b**) locating urban bottleneck areas.

2. Spatial Abstraction

Advanced sensing technology has collected a large amount of vehicle traffic data in transportation networks. Spatial abstraction refers to simplifying spatial data or transforming them into other forms of related variants for geospatial locations at different spatial scales, deriving abstract networks that consist of cells (regional compartments) and links. In such a mesh-data structure, the original trajectory data are aggregated spatially by nodes and network links and temporally by time intervals [54]. Figure 20a shows an abstract transportation network in Milan, Italy as established by Andrienko et al. [55] using GPS vehicle-trajectory data. Figure 20b shows vehicle flows between cells targeting the same traffic-flow data, demonstrating a certain inverse relationship between the traffic volume and the average speed. Furthermore, the same relationships as in a physical network exist in spatially abstracted networks; the fundamental diagram in the abstracted network refers to links of a physical transportation network (i.e., street segments), to which the relationships that exist in the spatial abstraction at different levels conform. Figure 20c shows traffic networks at different spatial scales, with the graphs on the right indicating interdependencies between relative traffic intensities and average speeds in the network links. This important discovery enables us to perform traffic analysis, prediction, and simulation in higher-scale spatial abstractions. The simulated trajectory in Figure 20d enables the analysis of link loads, the attained mean speeds, and the numbers of suspended cars in the cells, revealing bottlenecks in the transportation infrastructure.



(**a**) (**b**)

(**c**) (**d**)

Figure 20. Related examples of abstract-space visualizations [55,56]: (**a**–**c**) related abstract transportation networks for Milan that involve spatially abstract networks or physical street networks; and (**d**) the predictive model for spatial abstraction.

3.6. Time Series

Time-series data visualization analyzes data characteristics from two aspects of time and data, which represents the regularity of trajectory data changing with time and expresses the temporal law of trajectory data distribution (see Figure 21). We enumerate several common visualization methods for time-series data:

1. Linear graphs: Linear graphs are simple to implement. When displaying time-series data by this graph, one of the coordinate axes is fixed as the temporal axis to indicate continuous time, while the other axis is used to represent the data value that corresponds to the time of the data point.
2. Stacking maps: A stacking map shows the cumulative variations for different categories of data. However, it exhibits a poor ability to compare different types of data and poor performance when processing data with negative values.
3. Animations: The strength of animation is that it enables users' perception of data changes in the temporal dimension. However, in a dynamic case, users have worse memory in an overview, which is not conducive to data comparisons. Therefore, we do not recommend animations in general time-series data visualization.
4. Horizon graphs: As first proposed by Saito et al. [57], horizon graphs solve the problem regarding how some visualization methods cannot indicate negative values.

5. Timelines: A timeline represents changing time by a horizontal time axis within the time range of the described data and is typically used to indicate narrative trajectory data. However, for a long temporal span and dense data points, the overall layout becomes confusing, thus affecting the visualization performance.

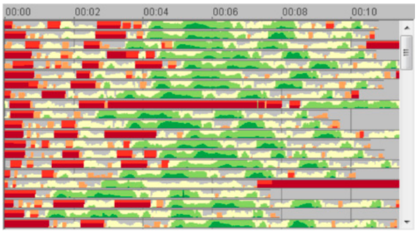


Figure 21. Time-series visualization of vehicle speed [38].

3.7. Transformation

1. Animal Collective Movement Trajectory Pattern

Thus far, the movement patterns of moving objects such as vehicles have been extensively studied, but few methods support collective movement analysis. Wood and Galton [58–61] proposed a general taxonomy of collective movement and a more specific taxonomy of spatial collectives. Compared to the former, the latter does not include the role structure and characteristics of intercommunications and interactions between individuals because relevant information is generally not available to animal collectives in movement data [62]. The positions of leaders in animal collectives may not be the front of a group. Therefore, Nagy et al. [63] employed a definition of "leading" close to "trend setting" [64]: when the movement direction of an object is imitated by another time-delayed object, a "leading" event occurs. The authors applied this "leading" to trajectories of a pigeon flock to select the time lag with maximal correlation. Figure 22 shows that hierarchical structure of collective flight could be more efficient than an egalitarian organization.

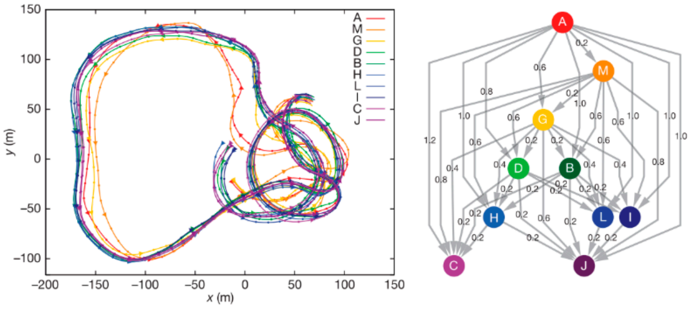
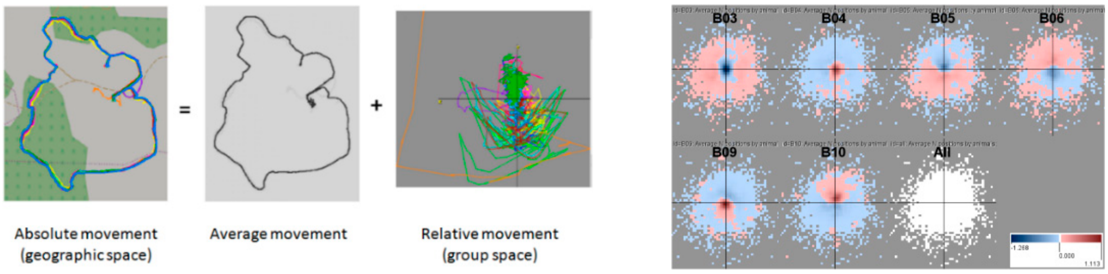


Figure 22. Hierarchical leadership network [63].

2. How to Implement?

When analyzing the movement of a collective, such as the movement of certain animals, we may examine which animals tend to wander to the front of the collective, whether the relative position of each animal changes over time, and other details. The related visual analytics methods are still primitive: they enable the extraction of collective-movement features, such as changes in the area and shape of the convex hulls over time in the collective, but cannot visualize these features well, failing to meet the analytical requirements of complex issues (e.g., animal-behavior analysis). In this regard, Andrienko et al. [65] proposed a method of spatial transformation visualization for the in-depth study of group movement. Figure 23a shows the foundation of spatial transformation: decomposing the absolute movement in geographic space into the average movement of the collective plus the relative movement of each individual in the collective space. A team of animal behaviorists from the University of Lethbridge and University of South Africa applied the spatial-transformation method alongside a density map to real movement data for 13 baboons in the De Hoop Nature Reserve in South Africa, yielding the relative baboon locations in their collective space [65]. Figure 23b shows the differences between the relative positional density of several certain baboons and that of all baboons. By examining the map series, we could recognize individuals that had significantly changed their positional tendency over time.



(**a**) (**b**)

Figure 23. (**a**) Fundamental idea; and (**b**) a baboon visualization example of spatial transformation [65].

4. Visualization Targeting High-Dimensional Data

Most people can hardly understand space with more than three dimensions; an insightful person may convert dimensions into a variable-resembled manner using common multivariate-data visualization techniques. However, two obvious problems arise, taking visual encoding in the icon method as an example: (1) despite a familiar visualization, the users' comprehension seems less convenient because the visual channels require being deciphered one by one; and (2) if the dimensionality grows continuously, visual channels seem no longer competent.

Low-dimensional trajectory visualization is not the only way to describe all trajectory data, although it can do so. Such methodologies inevitably trigger visual burdens of high-dimensional data in the era of information explosion, of which a possible consequence is over-plotting of the displayed data. We review the optimized visualization techniques of high-dimensional data for examples below, of which the extended higher data dimensions are an exciting topic for current and future research. Note that such constructions typically exist in a non-geographic space, and the conceptual operations of multidimensional and multivariate are no longer regarded separately. Instead, variables are converted into dimensions to continue exploration. After all, it is never easier to conceive a hypercube than a cube, and the simplicity of the process and results remain further considerations.

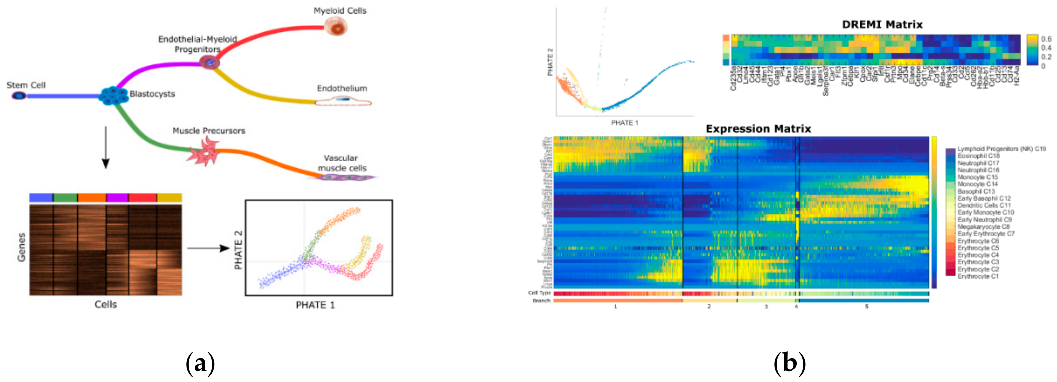
4.1. Dimensionality Reduction

1. Mapping Types

Generally, the analyzed trajectory datasets are of high dimensionality, and each data instance (point) is characterized by multiple features [66,67]. To visualize these data in 2D or 3D space, the dimensionality must be reduced to two or three. However, one issue of trajectory-visualization systems is that some dimensionality-reduction techniques cannot process massive trajectory datasets. Therefore, researchers have divided dimensionality-reduction mappings into two categories: linear dimensionality-reduction techniques (e.g., principal component analysis) and nonlinear dimensionality-reduction techniques (e.g., self-organizing maps and isometric mapping). The fundamental approach includes feature selection [68] and feature extraction [69]. Feature selection involves dimensionality-reduction mapping via selecting attributes of 1–3 dimensions (called dominant dimensions), while feature extraction is applicable to datasets without evident dominant dimensions. The multidimensional attributes of the trajectory dataset are reorganized to construct dominant dimensions and then implemented into the dimensionality-reduction mapping. Linear methods have high computational efficiency and relatively simple algorithms, so various linear methods [70] have been proposed to preserve different data features of the trajectories of interest. However, these approaches might miss nonlinear structures. Multidimensional scaling is a widely used dimensionality-reduction method to tackle nonlinear issues, which attempts to preserve differences between high-dimensional data points in a low-dimensional space. However, this approach usually fails to deal with massive points and requires much computational time [71].

2. PHATE (Potential of Heat-diffusion for Affinity-based Trajectory Embedding)

In recent years, dimensionality-reduction methods have become crucial for the interpretation, exploration, and visualization of high-throughput, high-dimensional biological data because these methods can extract the main trends in the data while discarding noise. However, biological data consist of a branching, usually nonlinear progression structure, which arises from underlying biological processes (e.g., differentiation and population drift). PHATE, proposed by Moon et al. [72] is a dimensionality-reduction method that visualizes trajectory structures in high-dimensional biological data and was explicitly designed for preserving this progression structure in data. PHATE also provides interpretable overlays, emphasizing and revealing trajectory end-points, branch points and associated split-decisions, progression-forming variables (e.g., specific genes), and paths between developmental events in cellular state-space. Figure 24a shows the embryoid body-differentiation process, visualizing different cell fates in terms of genes along trajectories, which is crucial to understanding what drives certain paths. Furthermore, the authors devised a DREMI (conditional-Density Resampled Estimate of Mutual Information) matrix to display mutual information that varies along the trajectory (see Figure 24b).



(**a**) (**b**)

Figure 24. PHATE embedding technique visualizing trajectory data in the biological domain [72]: (**a**) embryoid body differentiation; and (**b**) DREMI matrix that displays trajectory-related details.

4.2. Projection

1. Projection Methods

Specifically, we first assume that high-dimensional trajectory data can consist of m dimensions, i.e., each data item includes *m* values. Each high-dimensional trajectory dataset may consist of *n* such data items. When $n > 3$, traditional approaches are unsatisfying in terms of perceiving and comprehending these data. In this case, we can select a certain projection method, projecting the N-dimensional data onto a 2D plane, so that users can intuitively grasp the data distribution. Although dimensionality-reduction techniques are generally combined with multidimensional projection (MP) in the visualization community, as are scatter plots and parallel coordinates, projection-visualization techniques are our interest. Thus, we introduced projection as an individual method. The next question is drawn by the projection method: typical projection methods consist of projective projection and affine projection, resulting in distortions of different degrees and manners (see Figure 25a).

Only affine projection based on orthographic projection maintains a stable relative distance. If the orthogonality of the projection is well preserved during the interaction, the advantage includes a form of high-dimensional data exploration that reduces the deviation and can find a better distribution for mappings of different dimensions through a data-tour strategy. However, the comprehension difficulty increases when applied to massive multidimensional trajectory data. Inverse projection, introduced by Elisa [73], creates new multidimensional data in a 2D environment, improving the projection quality with slight control and entirely removing the complexity of dealing with multiple dimensions (see Figure 25b).

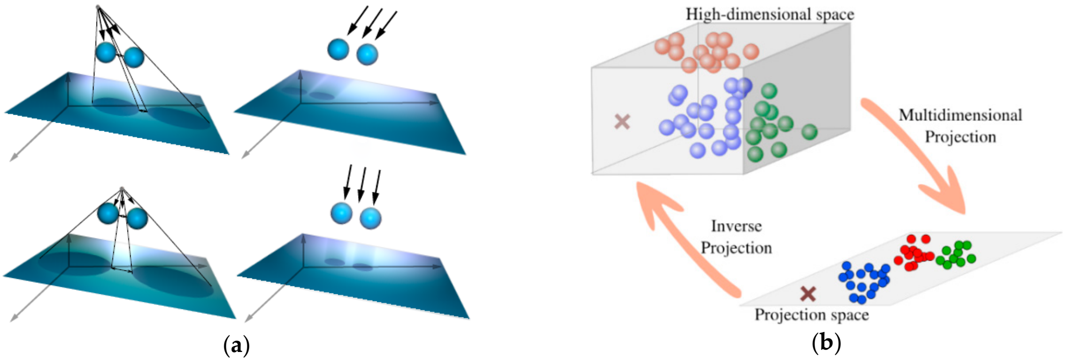


Figure 25. (**a**) Various projection modes [74]; and (**b**) inverse projection workflow [73].

2. Density Projection

In the field of multidimensional trajectory visualization, rendering large sets of lines is a relevant topic, especially the visualization of integral curves in vector fields. Geometric visualization often suffers from visual clutters because of the growth of mutually occluded curves and the potential to conceal related features. Attempting to avoid this effect often requires complex data processing for the seeding, clustering, or filtering of related lines. Kuhn achieved density projection via an additive or subtractive color-blending scheme, emphasizing areas with increased trajectory density. Figure 26 shows the ocean-flow results on climate-model simulations, where density-projection rendering enabled a comparative view across dense trajectory sets.

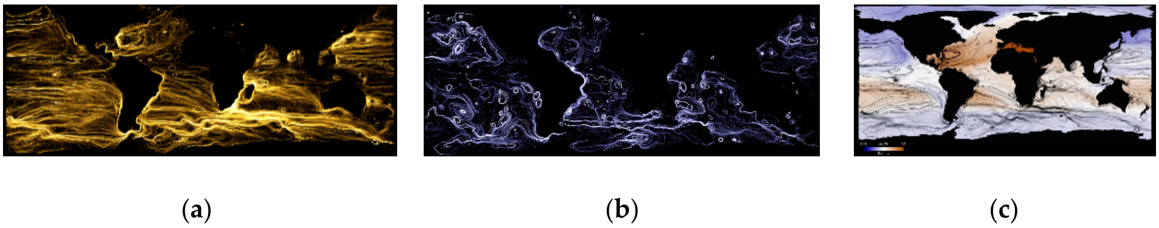


Figure 26. Additive projection of streamlines from ocean simulations: (**a**) at depths from 0 m to 500 m; (**b**) at depths from 2500 m to 3000 m; and (**c**) at 100 m with the salinity field [75].

3. Dynamically Interactive Visualization

Currently, MP methods are transforming from static displays to more interactive displays, enabling human intervention to modify the layout and facilitate data exploration and comprehension. Elzen et al. [76] proposed a projection-based method to analyze the change patterns of dynamic graphs, converting each time-step network into a high-dimensional vector, which was then projected onto a two-dimensional plane. Figure 27 shows that each point in the projection layout on the right indicates a snapshot of a time step. This approach can effectively help the user explore the steady state, recurring state, and abnormal state of the dynamic graph and the transition process between these states.

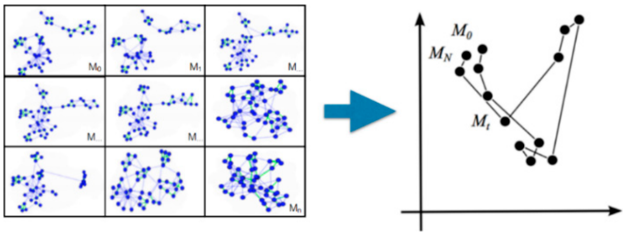
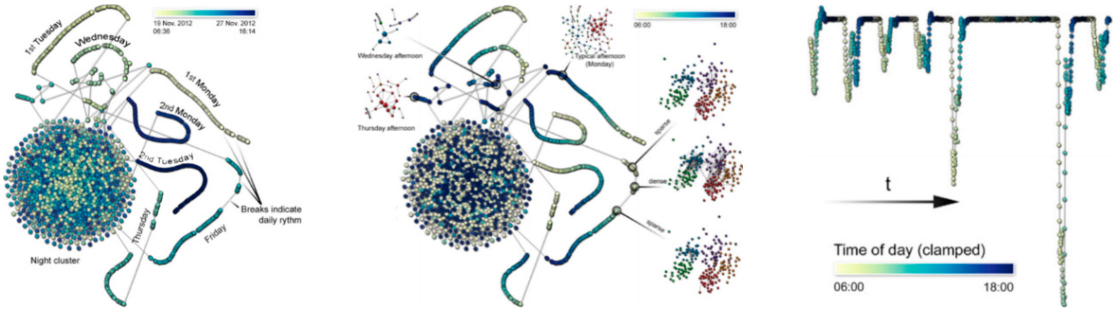


Figure 27. Dimensionality-reduction process that transforms the snapshot on the left into two-dimensional points on the right [76].

During the projection process, each snapshot must be converted into a high-dimensional vector. Figure 28 shows the weekday dynamic trajectory data recorded by real data. After preprocessing, the resulting dynamic graph network had a total of 180 vertices, 10.104 edges, and 2015 snapshots. Figure 28a,b shows the visualized views from a nonlinear projection method, with different periods encoded in different colors. Figure 28c shows the result of replacing the horizontal axis with a time axis in the PCA projection.



(**a**) (**b**) (**c**)

Figure 28. Dynamic trajectory-data visualization of high-dimensional projections from analyzing high-school contact patterns [76]: (**a**,**b**) temporal-information maps in different periods; and (**c**) horizontal axis map over a certain time.

4.3. Hierarchy

1. Hierarchy in Multidimensional Data

The fundamental idea of hierarchy-based multidimensional trajectory data-visualization techniques involves dividing the multidimensional space into multiple subspaces, organizing these subspaces in a hierarchical structure, and finally integrating and visualizing them within a view. For example, Liu et al. [77] built a network that comprised all taxi trips in a dataset of taxi trajectories in Shanghai, clearly depicting a two-level hierarchical structure (see Figure 29). This study confirmed that short journeys dominated local spatial interactions, while long trips served to connect these local clusters instead of following the same short-journey pattern and maintaining the enhancement process of subareas. Common hierarchy-visualization techniques include node-connected, space-filled, and three-dimensional forms. For example, radial space-filling (RSF) hierarchy visualization has the advantage of effectively utilizing the display space while conveying the hierarchical structure in the view [78]. From the perspective of legibility, this approach is a highly demanding multidimensional data-visualization technique.

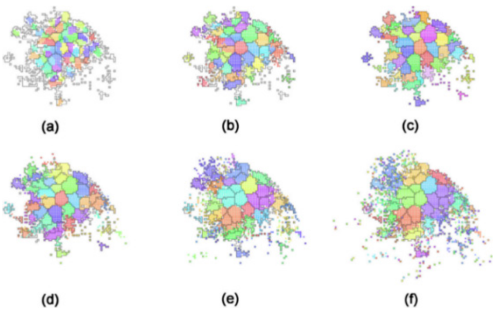
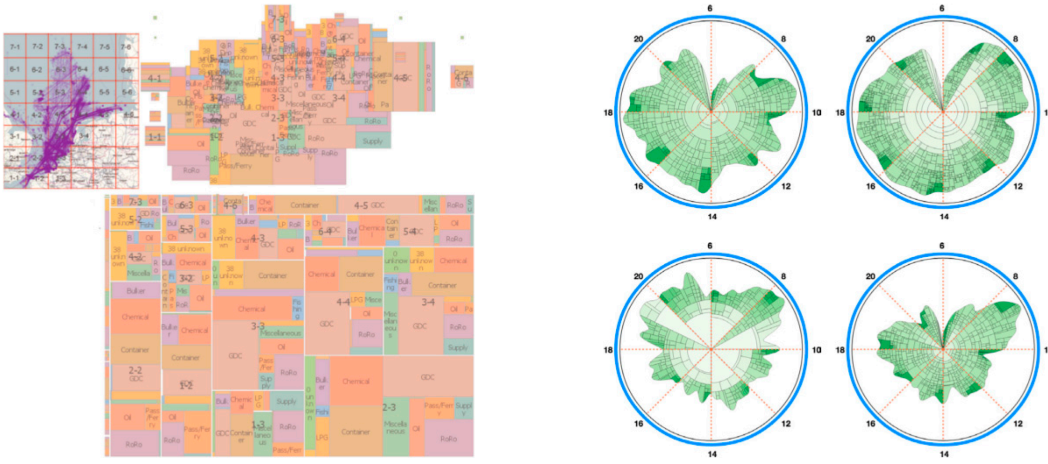


Figure 29. Visualizations of cluster detecting results [77].

2. Treemaps

Treemap was originally proposed by Shneiderman et al. [79] and is a popular visualization method that solves the problem of spatial constraints by using a hierarchical structure. This method represents the overall data as a large rectangle, in which a small one with a certain area indicates the individual data, and the relationship between each individual is shown as the inclusion relationship of the spatial position of the small rectangle. Wood et al. [80] proposed a spatially ordered treemap (SOT) technique that considers the geolocation of nodes (see Andrienko et al.'s application in Figure 30a [40]). Such treemaps aggregated with space and attributes have a rectangle-based space-filling hierarchical layout. Several common implementations of treemaps also exist. When visualizing location features simultaneously in different periods based on a radial layout, a circular treemap tends to waste space and undermine the efficacy of the radial design. Sunburst trees [81] divide the radial space into several segments, each of which indicates a period, and then aggregate information for each period and display them in various segments. The main drawback of this design is the challenge of selecting the length of each period. Therefore, Wu et al. [82] proposed a contour-based treemap to map the spatiotemporal distribution of corresponding movement data, with the shapes of contours and the colors of contained sectors indicated. To obtain insight into the causes of the co-occurrence intensity of each region in the city, the authors used a contour-based treemap, TelCoVis system, to analyze the telco data over a time span in representative regions of different functions selected from Guangzhou, enabling users to explore the spatiotemporal characteristics of human mobility in areas where co-occurrence occurs (see Figure 30b).



(**a**) (**b**)

Figure 30. (**a**) Spatially ordered treemap visualization [40]; and (**b**) TelCoVis [82] visualization results that display human mobility in different types of sites.

3. Movement-Trajectory Capture

Hierarchical methods can use existing movement-trajectory sequences for motion synthesis. However, browsing and analyzing such data or extracting key motion sequences can be difficult because of the enormous data amounts and the high-dimensional and time-sequence features. Bernard et al. [83] developed a MotionExplorer system to view and analyze functions in the visualization method of movement-trajectory data capture, whose interface is presented in Figure 31. After acquiring the human movement-trajectory data, the system identifies the specific coordinates of each marker point, measuring the motion similarity.

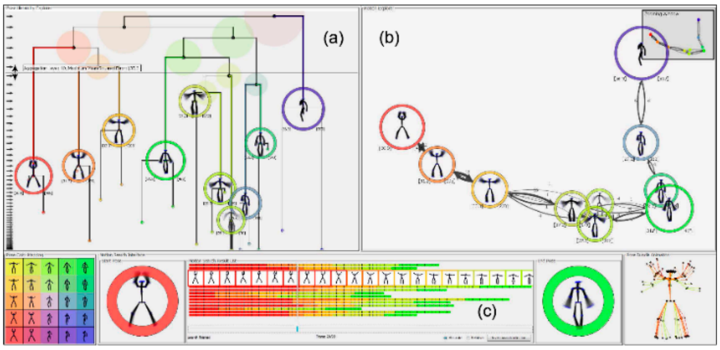


Figure 31. Snapshot of MotionExplorer interface with: (**a**) a treemap for browsing the hierarchical structure; (**b**) a node-link diagram viewing motion aggregations and transformations; and (**c**) a search view for query and result analysis [83].

4.4. Pixmaps

Similar to hierarchy-based techniques, the basic idea of pixel-oriented multidimensional visualization techniques is dividing high-dimensional space into multiple subpanes according to the dimensionality of multidimensional data. However, the distinction is that each pane corresponds to one dimension of the multidimensional dataset and treats each pixel as a multidimensional data point, which is colored according to different attribute values in individual panes. The position arrangement of pixels, according to various demands, can be either radially distributed from the center according to the correlation between data item and a certain attribute or arranged row by row or column by column based on a certain attribute. Here, we introduce circular pixmaps and columnar pixmaps, which are commonly used in trajectory visualizations.

1. Circular Pixmaps

Circular pixmaps arrange multidimensional data in different sectors from center to circumference in units of pixels, and the overall circle that comprises these sectors represents the entire high-dimensional space. For example, establishing an association between some decision-making events in a city to its OD pattern variations by using matrix decomposition. Circular pixmaps are a novel visual encoding of the projected trajectory data space, which involves the center of the selected region as the coordinate origin and the direction and distance as coordinate axes, into a discrete polar coordinate form (see Figure 32).

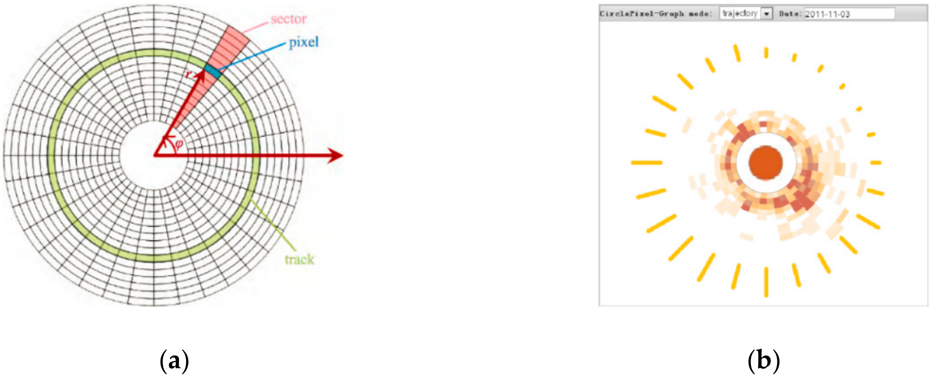


Figure 32. Circular pixmap design: (**a**) a prototype of circular pixmap; and (**b**) an example of circular pixmap [84].

2. Columnar Pixmaps

The above temporal granularity is based on day-by-day analysis, so the analyst may further explore the trajectory of interest between the starting areas and the ending areas. The three trajectory attributes of time range, road network length and travel duration are of major importance for studying the characteristics, taxi routes and traffic conditions of people's activities. The columnar pixmap in Figure 33 visually represents the complex relationship between these three attributes, where each embedded pixmap is distributed in segmentations based on a grid. By segmenting the trajectory speed or travel duration into continuous intervals, each pixel bar within the grid visualizes the speed or duration of the spatiotemporal trajectory over a particular time and a distance range.

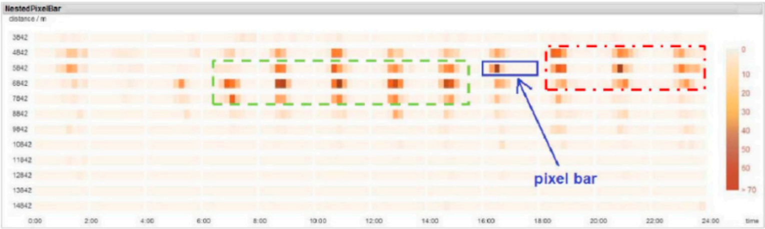


Figure 33. Columnar pixmap design [84].

3. Availability

In light of the efficacy of circular and columnar pixmaps in mining the space–time and attribute information of trajectory data, Wang et al. [85] designed a pixel-based tabular visualization for the traffic speed and road congestion events in Beijing (see Figure 34a). This approach provided a compact view to enable displaying multiple roads side by side for comparison and extracted daily and weekly speed patterns in the visualization. In addition, for the traffic issue, Zheng [84] used a pixel method to show the distribution of taxi trajectories 24 h a day in Huanglong Stadium in China (see Figure 34b). Two situations of interest can be observed in Figure 34b: first, the route to the airport and station during the day had more requirements for taxi services; and, second, the terminal distribution at night was exactly the opposite, and both the taxi demand and distance increased. The reason for this abnormal pattern could be the existence of nightly football matches and concerts at these locations, hence a large demand for taxis. Therefore, traffic management can take advantage of pixel methods of trajectory visualization to analyze crowd travel patterns and provide a reference for urban traffic planning.

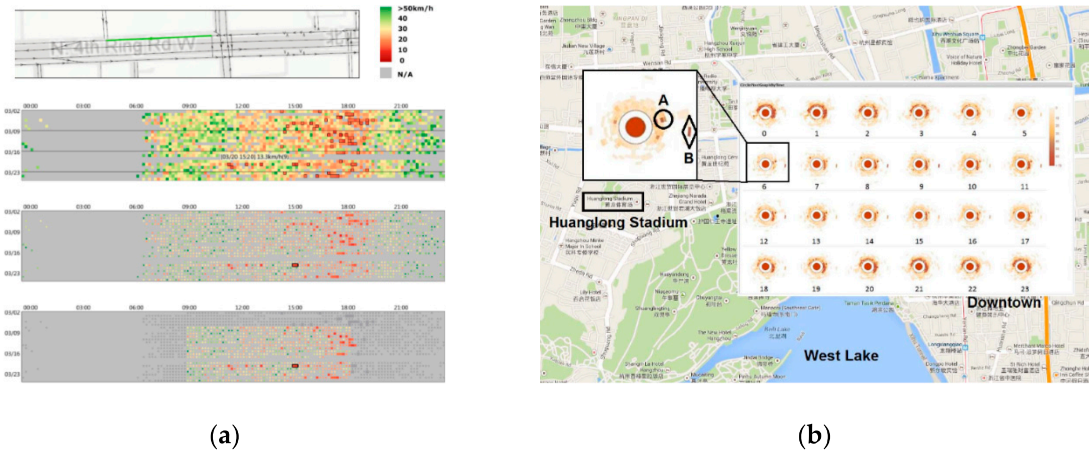


Figure 34. Examples of pixmap visualization: (**a**) tabular pixmaps for speed-pattern exploration [84]; and (**b**) 24-hour taxi-trajectory patterns [85].

4.5. Parallel Coordinates

A parallel coordinate system mainly solves the problem that expressing massive, multidimensional and complex information seems troublesome in traditional two-dimensional graph representation.

1. Diversity of Research

Parallel coordinate plots (PCP) represent one of the most advantageous techniques for spatial visualization and the exploration of high-dimensional trajectory attributes. Their fundamentals are representing N-dimensional attributes of multidimensional data in the form of N parallel axes of equidistant verticals, each of which is mapped to an attribute dimension. The origin of parallel coordinates as a multidimensional geometric visualization method can be traced back to early work by Inselberg [86] and Wegman [87], which was extensively and mathematically described afterward [88]. These techniques can provide static results through breaking up high-dimensional space into multiple pre-selected low-dimensional spaces. Itoh et al. [89] interactively built a set of low-dimensional PCPs through sampling the subsets of high-dimensional data spaces that were selected by the user and, visualizing the most meaningful dimensions. Figure 35a shows a snapshot of this technique, which enables the extraction of more complex multivariate relationships than pairwise correlation. However, these methods cannot effectively visualize subsets or subspaces of high-dimensional space without an interaction mechanism for selecting dimension sets. Guo et al. [90] calculated a spatial interaction indicator and used a parallel-coordinate method to visualize taxi data changes in different regions in Shenzhen (see Figure 35b).

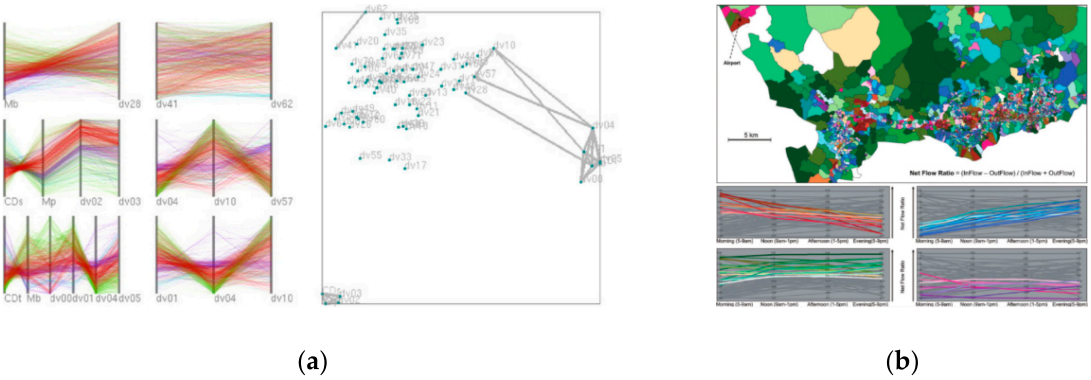
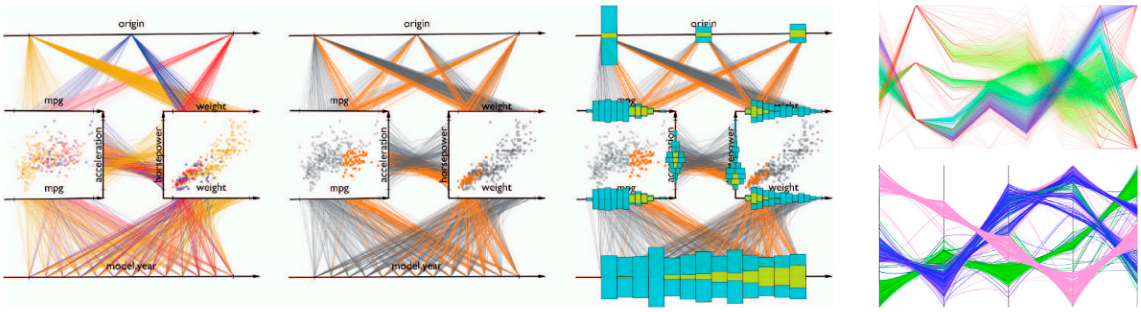


Figure 35. Parallel-coordinate visualizations: (**a**) low-dimensional PCPs and a corresponding dimension graph [89]; and (**b**) PCP views of variations in the number of taxis [90].

To improve the performance of correlation analysis, several hybrid methods can combine parallel coordinates with other visual mappings. For example, small scatter plots can be placed on top of parallel coordinates [91] or embedded between adjacent axes of parallel coordinates [92]. Claessen et al. [93] designed FLINAPlots, which aggregates scatter plots and histograms onto parallel coordinates and enables analysis with various visualizations from multiple perspectives (see Figure 36a). In the context of trajectory big data, major issues of parallel coordinates include line cluttering and overlapping from massive multiattribute trajectory data items. A solution to these problems is simplifying parallel-coordinate plots according to the line-aggregation characteristics, yielding a cluster visualization [94,95] (see Figure 36b).

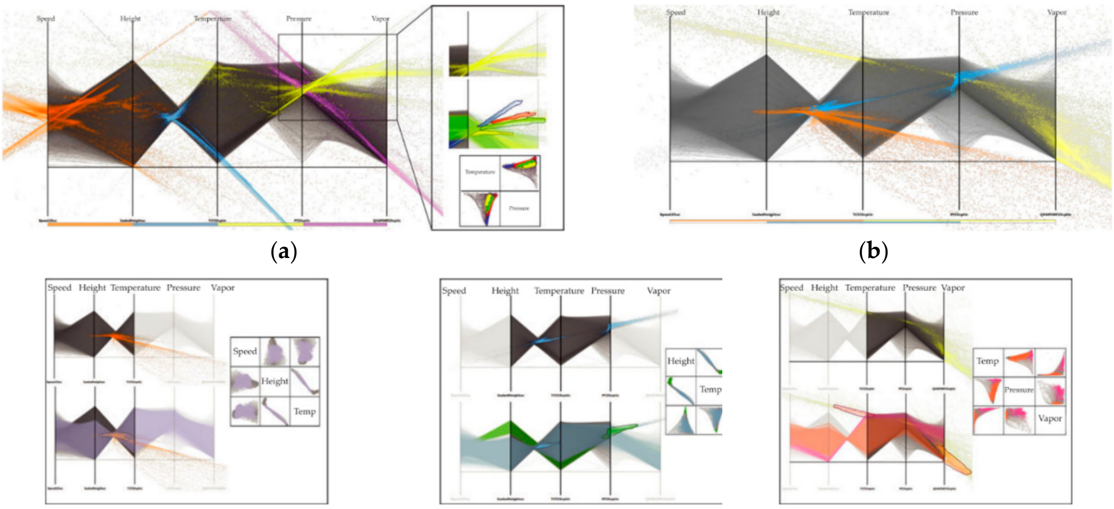


(**a**) (**b**)

Figure 36. (**a**) FLINAPlots visualization integrating scatter plots and histograms [93]; and (**b**) aggregation for parallel coordinates [94,95].

2. Case Study with a Complex Dataset

Unlike simple geometric shapes, a trajectory dataset of Hurricane Isabel involves much complexity, with more local linear patterns observed in the data domain. Figure 37 is a linear fitting based on the local neighborhood of the data points, creating a multiattribute data field for the hurricane trajectory. Generally, the cluster is associated with the area at the bottom of the data around the eye of the hurricane.



(**a**) (**b**)

(**c**) (**d**) (**e**)

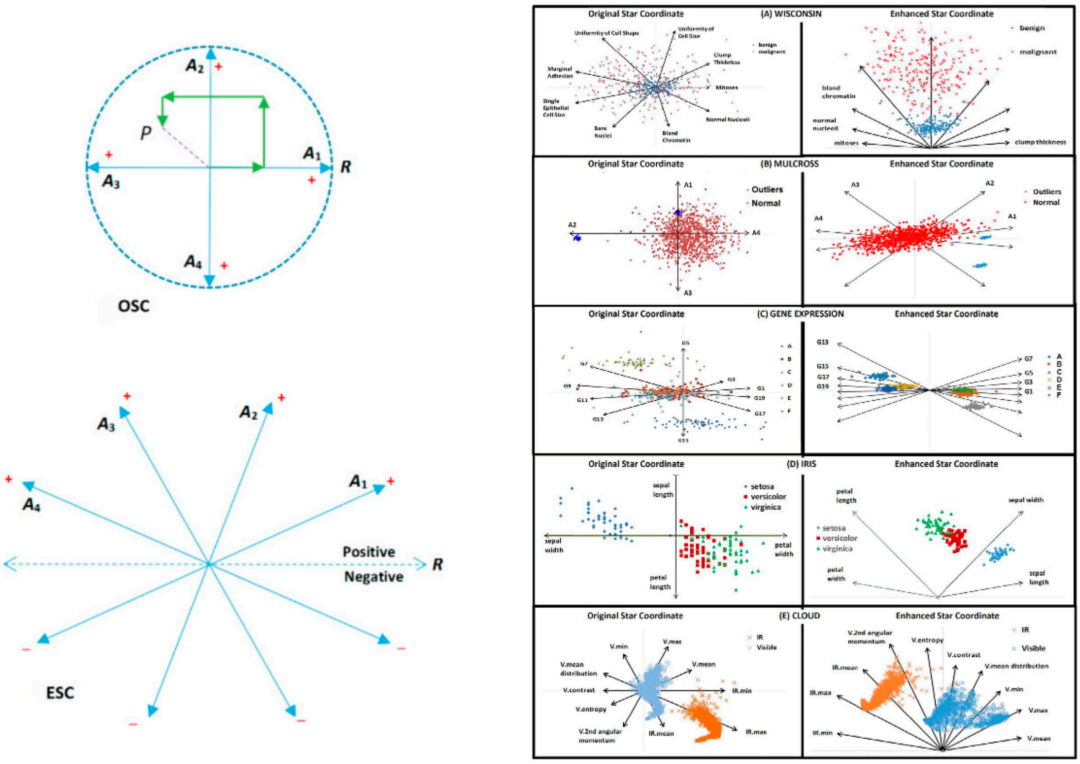
Figure 37. Visualizations of Hurricane Isabel's trajectory dataset with five attributes: (**a**) the analysis result of one-flats, indicating positive and negative correlations between attributes; and (**b**–**e**) two-flats and their analysis [96].

4.6. Radial Graph

1. Star Coordinates (SC)

We consider star coordinates [97] as a representative of those methods that depicts an N-dimensional data space with a two-dimensional radial graph. Such a plot has n axes (i.e., A1, A2, . . . , An) that represent different dimensions in the dataset. All these axes are radial rays from the origin, with each axis Ai inclined at an angle of 2(i−1)π/n with A1. Through a multidimensional interactive arrangement, star coordinates can help users discover clusters in hierarchically defined datasets, which can always be preserved as a point cloud in the visual space via linear mappings. However, when these mapped point clouds overlap each other, their boundaries become obscured and indistinguishable. At this point, users can push, pull or rotate the axes until the desired result is achieved. Fundamentally, techniques such as star coordinates are variants of parallel coordinates, where the axes of each dimension share a common origin [97].

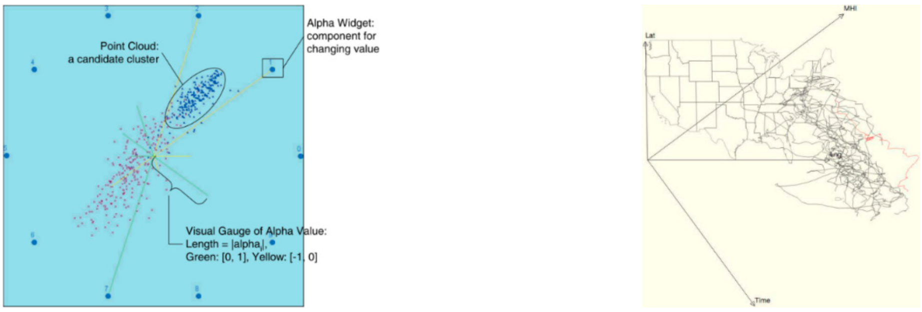
Original star coordinates (OSC) are intended to display multidimensional data in a two-dimensional visualization space. A natural extension is to stretch the visualization space to three dimensions [98], which expands the data-exploration space and helps unearth subtle patterns hidden in a two-dimensional space. However, two of the flaws before expansion still exist: the non-preservation of original data signs and the exact opposite placement of independent axes. Regarding both shortcomings, enhanced star coordinates (ESC) exhibit better performance with all datasets compared to OSCs. Figure 38a shows a common model of OSCs and ESCs, and Figure 38b shows visualization results from five datasets. Obviously, ESCs can distinguish spatially separated clusters more clearly than OSCs.



(**a**) (**b**)

Figure 38. (**a**) Original star coordinate system with a sample of point P (top) and enhanced star coordinate system (bottom) [99]; and (**b**) comparisons of visualization results between original star coordinates and enhanced star coordinates in an example dataset [100].

An interesting application of ESCs is the VISTA system [101], which applies linear mapping to avoid breaking clusters after mapping from a K-dimensional space to a two-dimensional space (see Figure 39a), whose visualization output can be used to confirm the validity of cluster structures. The major disadvantage of this method is that it only satisfies the visualization of K-dimensional data with K < 50. Thus far, several extensions have been introduced in VISTA, which aggregate static clusters for low-dimensional data. Regarding the fact that the substantive analysis of trajectory data could involve variables that exceed space and time, Murray et al. [102] proposed an interactive positioning method based on star-coordinate extensions specifically for multidimensional spatiotemporal trajectory data, namely StretchPlot, which enables users to map the trajectory-set variables into a high-dimensional space and represent them as a connected linear sequence. This approach nests sequential events (with associated variables) within entities, joining them according to their chronological order to form trajectories. Figure 39b demonstrates this technique using trajectories linked with performance dates and geographic locations, where the various multidimensional relationships among genre, time, popularity and geography are explored.



(**a**) (**b**)

Figure 39. Enhanced star-map visualization: (**a**) star-coordinate system implementation with the VISTA model [103]; and (**b**) StretchPlot abstracts the trajectories from geospatial movements into abstractive movements of demographic and social variables [102].

2. RadViz

RadViz is essentially a radial-projection type of multidimensional geometric-mapping technique. This approach fixes dimensions on a circular view as dimensional anchors; all data points are plotted to positions where the resultant force is zero under all spring forces. The magnitude of spring forces is proportional to the data values in corresponding dimensions, thus data points with similar dimension values are clustered into close positions. With the fine interactivity, scalability, and retentive ability of original features, RadViz adequately analyzes the attribute values of moving objects. In the comprehensive analytic system for trajectory and transaction in Section 2.2 [10], Zhao et al. [104] also used RadViz to address the uncertainty of employee-location types, assisting in depicting individual living habits and detecting suspects' abnormal behaviors (see Figure 40).

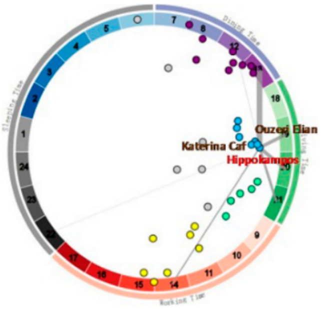


Figure 40. Identifying location types in digital-forensics process via Radviz [104].

3. Radar Charts

Radar chart (or spider chart) is also a radial-projection type of multidimensional geometric mapping method. The chart emits axes outward starting from the same point, projecting three or more variables in a multidimensional space into a two-dimensional space and connecting projected points for each variable to generate a closed polygon as a data record in multidimensional dataset. The number of polygons in a radar chart is limited: if there are N objects to be evaluated, overlapping and cluttering could occur in all aspects of contours, charts, or filled areas, making the data hard to read. Therefore, we should keep the simplicity of radar charts and limit the variable counts. Additionally, despite the reference to the grids, visually comparing the variable values is still uneasy because of the uncertain radial distance. Figure 41 shows our simulated exploration of attribute distributions along major course of several vessels in a certain domain.

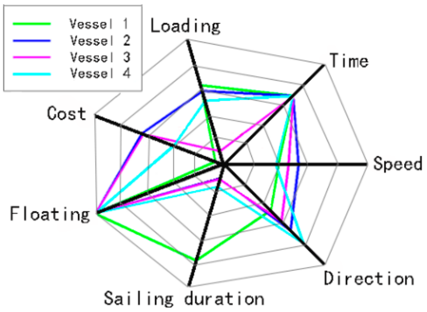


Figure 41. Radar chart of vessel attributes.

5. Universal Multidimensional Visualization

In this section, we introduce relevant visualization techniques that potentially fit either low- or high-dimensional trajectory data types, or that function in using these trajectory data to describe visualization results of user tasks instead of being dependent on the data types. Note that choosing a rational approach depends on many factors and almost always involves trade-offs.

5.1. Clustering

1. Spatial Cluster Separation

Clustering (or cluster analysis) refers to aggregating similar trajectories into different subsets, each of which is defined as a cluster. The rule is that the objects in the cluster are highly similar, whereas those between the clusters are not similar. Differently from static data in traditional algorithms, trajectory data show the characteristics of serialization, triviality and redundancy. Thus, many trajectory clustering studies focus on extending well-known traditional clustering algorithms. Take vector field k-means method as an example: Figure 42a shows that Ferreira et al. [105] proposed a vector field k-means algorithm based on vector-field fitting, fitting hurricane trajectory data to the vector field model to explore the movement pattern implied by the data. The progressive clustering method for clustering flight trajectories has also been developed in the field of air-traffic research [106,107], which applies clusters with different distance functions or parameter settings to various data subsets. Andrienko et al. [108] applied this algorithm to a flight-trajectory study in Spanish airspace, enabling analysts to extract the main routes within upper and lower subspaces of Spain and providing the foundation for air-navigation services (see Figure 42b). Additionally, this method can process unevenly distributed data. For example, Rinzivillo et al. [109] used progressive clustering to detect coherent clusters in data with uneven spatial density when depicting vehicle trajectories towards workplaces on weekdays. Figure 42c shows the results of three steps of progressive clustering according to trip destinations.

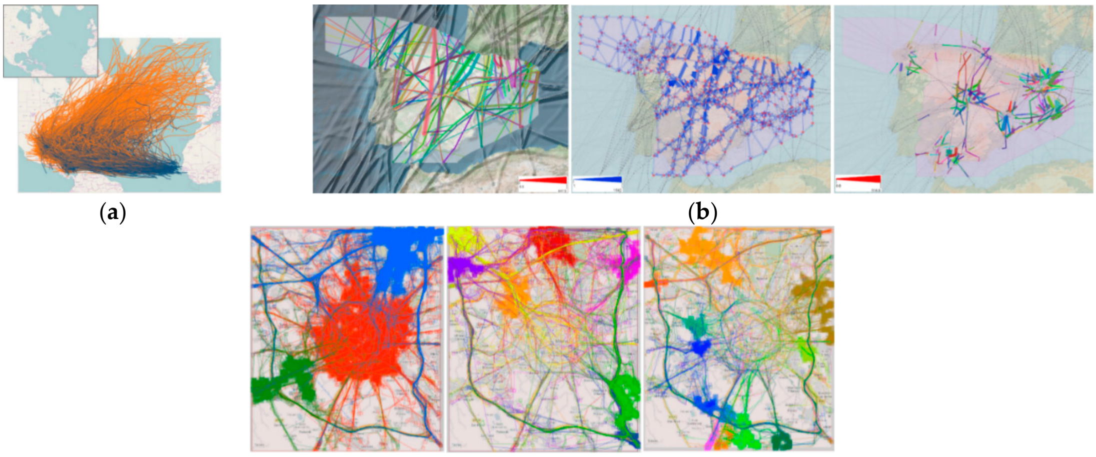
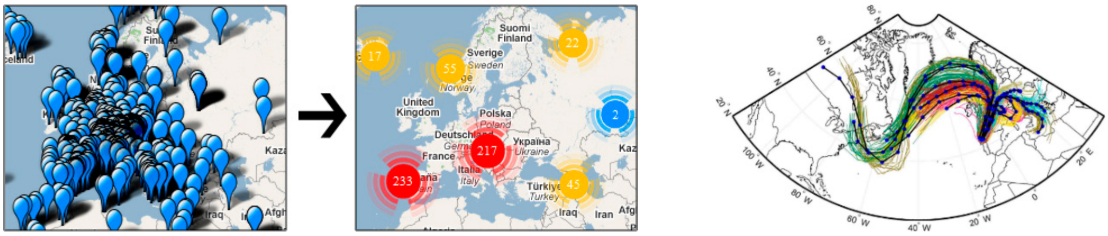

(**a**) (**b**)
(**c**)

Figure 42. Visualization examples of spatial cluster separation: (**a**) reconstruction of a generalized air traffic network by selecting and clustering relevant parts of flight trajectories [105]; (**b**) visualization results of vector field k-means clustering hurricane trajectories [108]; and (**c**) top-down hierarchical compartmentalization of the original dataset by progressive clustering, where the largest clusters are hierarchically and gradually removed to reveal secondary clustering results [109].

2. Clustering Structure

When a plethora of markers are displayed on the map, displaying all individual elements is no longer necessary and would cause problems such as poor display-and-interaction performance and visual overload. In this case, point-element overlapping and low loading speed may occur. Even without overlapping, the implied information cannot be captured by our naked eyes while visualizing hundreds of points simultaneously. The solution to this problem is to aggregate multiple sets of point features into one symbol, i.e., a cluster. The fundamentals of a cluster graph are aggregating statistical data into a small area and visualizing them so that the screen size affects the aggregation process. Figure 43a shows a cluster map obtained after processing around 600 point elements. In addition to point elements, Ramos et al. [110] solved the problems of line cluttering and overlapping when horizontally separating different air streams and rendered the clustered centroid trajectories (see Figure 43b).



(**a**) (**b**)

Figure 43. Clustering visualizations of different types of elements: (**a**) comparison of point elements before and after clustering [111]; and (**b**) clustered result of line elements [110].

3. Bundling Techniques

Bundling techniques enable effective displays of the coarse-scale connectivity structures of trajectory sets that link a set of spatial locations in a clutter-free manner [112–116]. For example, Hurter et al. [117] applied the efficient and robust kernel density bundling (KDEEB) method [112] to a streaming graph that contained only trajectories whose start time fell within a sliding time window to display how flight patterns changed in a geographical area over a week. However, their solution did not derive any other attributes of the emerging bundle, such as the flight direction, altitude or speed. Klein et al. [118] extended this approach by combining bundling with animations based on multivariable attributes. The authors projected the attribute values of corresponding sampling points on the bundled trajectories, creating a multivariate animated visualization. However, when the trajectory data become huge, the visualization effect could be affected by visual clutters. The focus + context interactive technique helps further reduce clutters and enable complex data queries in trajectory visualization. Figure 44 shows aircraft trajectories as visualized by various bundling techniques.

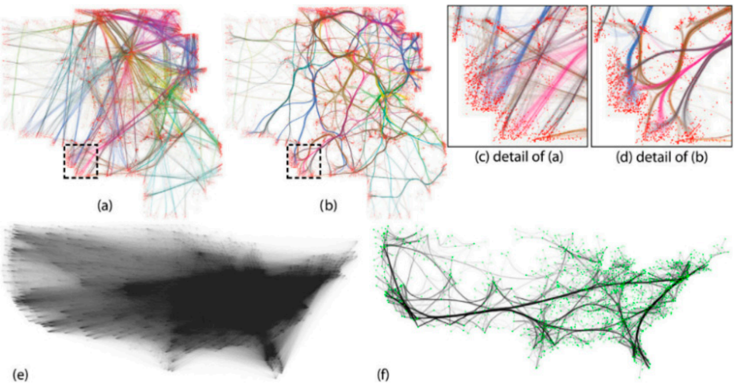


Figure 44. Bundled-trajectory visualization [119]: (**a**,**e**) unbundled trajectory sets; (**b**) the bundling of (**a**) with the SBEB method; (**f**) the bundling of (**e**) with the KDEEB method; and (**c**,**d**) close-up views of (**a**,**b**) (image courtesy of Hurter et al.).

4. MobilityGraphs

Mass-mobility visualization methods that imply location features can only represent the pattern of spatial interactions but reveal no spatial distributions, hence losing the spatial pattern. Analyzing mobility datasets requires a comparison of spatial circumstances (i.e., presence and flow of people at certain times) and the comprehension of spatiotemporal variations. Conventional flow visualizations often fail because of excessive clutter. The MobilityGraphs visualization by Von et al. [120] abstracts regions into nodes by combining spatial and temporal simplifications; the interaction between these nodes is the connecting edge. Figure 45 shows an experiment on an aggregated dataset derived from both geo-tagged Twitter posts and mobile phone call data records in London, visualizing the spatiotemporal variations in movements in a long time series and enabling the recognition of the daily and weekly patterns of residents' movements.

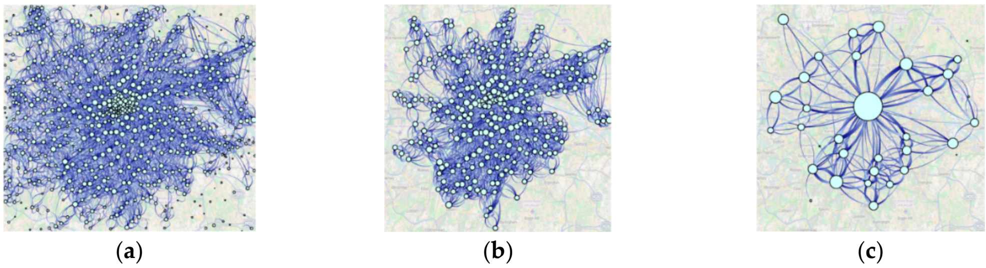
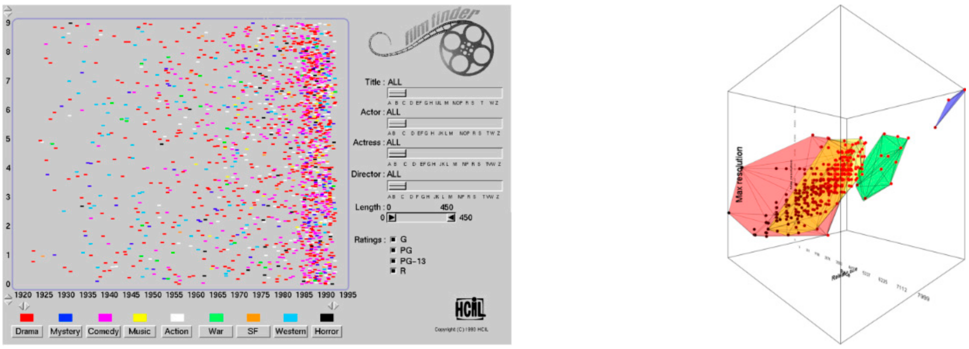


Figure 45. Visualization of London resident-mobility data [120]: (**a**) original flows; (**b**) filtered flows are cluttered; and (**c**) aggregated flows revealing the spatial patterns.

5.2. Scatter Plots

1. Generalized Scatter Plots

A scatter plot [121] is a type of plot or mathematical diagram that displays a set of data values (typically two variables) with Cartesian coordinates. If the points are color-coded, an additional variable can be displayed. The data are displayed as a collection of points, each of which has variable values that correspond to the dimensions [41]. A two-dimensional scatter plot maps attribute-value sets of two dimensions in multidimensional data onto two axes while reflecting the attribute values of other dimensions by various visual elements of the graphic marks in the plane as determined by the two axes (see Figure 46a). However, these elements are distributed over two dimensions, so the exploration of additional dimensional relationships or connections is relatively weak. Thus, Elmqvist et al. [122] extended this method to a three-dimensional space (see Figure 46b).

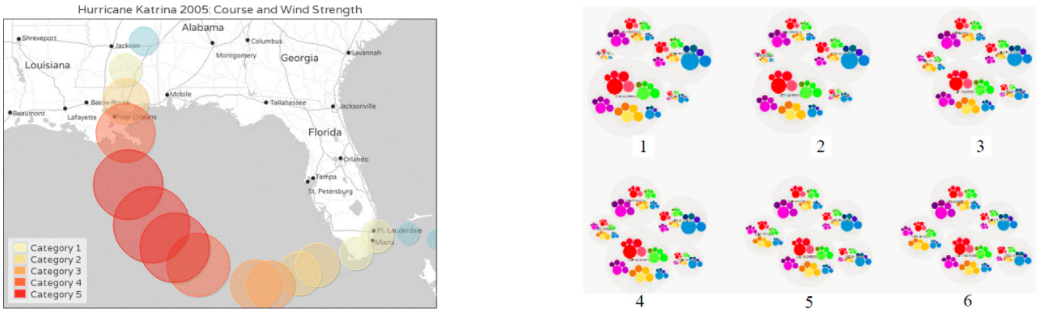


(a) (b)

Figure 46. Scatter-plot visualizations: (**a**) 2D scatter plot [123]; and (**b**) 3D scatter plot [122].

2. Bubble Charts

As a variant of scatter plots, bubble charts indicate the third dimensional data with data-point sizes. As stated in the documentation for Microsoft Office, "You can use a bubble chart if your data has three data series that each contain a set of values. The sizes of the bubbles are determined by the values in the third data series." [124] Figure 47a expresses multidimensional trajectory events of Hurricane Katrina in terms of geographic latitude and wind intensity. Moreover, the combination of trajectory-aggregation information and other dimensional information can display the trajectory thematic features in a bubble-visualization view. If a temporal dimension is added, the trajectory-change trend is better depicted. The bubble charts in Figure 47b dynamically present the user-clustering results calculated by selected topic model as smooth transitions [125].

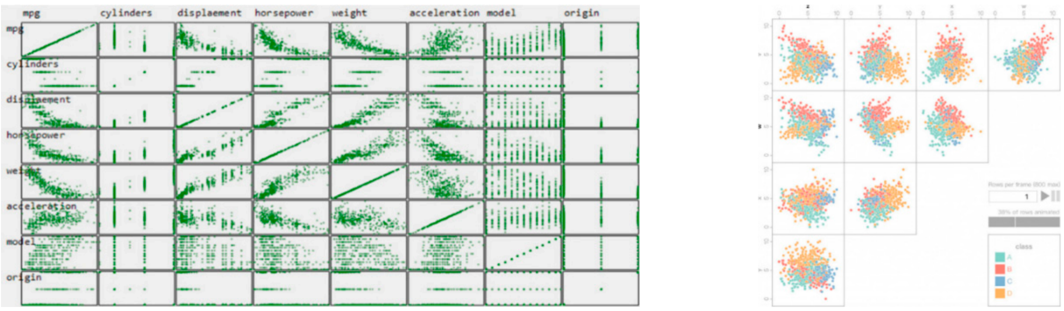


(a) (b)

Figure 47. Examples of bubble charts: (**a**) a trajectory-event visualization; and (**b**) temporal transformation results.

3. Scatter Plot Matrix (SPLOM) and Dynamically Interactive Visualization

SPLOM [122], proposed by Elmqvist et al. from INRIA, France, is derived from the idea of scatter approach in the domain of multidimensional data visualization. The scatter plot matrix in Figure 48a shows all the pairwise scatter plots of attributes on a single view, with multiple scatter plots in a matrix format [126], as the result of dimensionality reduction. However, Multiclass SPLOMs also have problems with over-plotting, and most existing techniques to counter this problem only focus on individual scatter plots with a single class. An animated SPLOM by Chen et al. [127] that uses a flicker-based animation exhibits better performance in identifying dense areas and is even more easily interpreted and more powerful than static scatter matrices. Figure 48b shows this animated approach on several typical multiclass and multivariable datasets of different scales (accessed from vgl.cs.usfca.edu/animated-sploms). The results demonstrate that animations mitigate over-plotting in this challenging multiclass setup.



(**a**) (**b**)

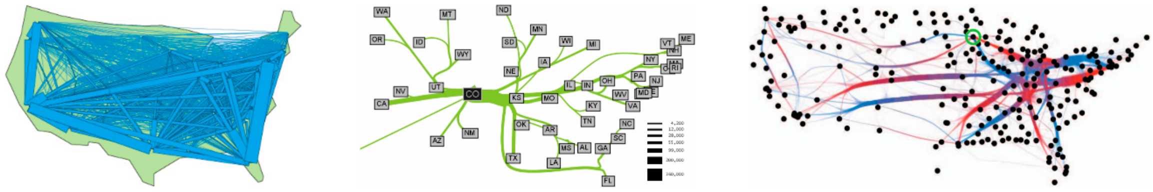
Figure 48. SPLOM visualization examples: (**a**) exploration of a seven-dimensional car dataset [126]; and (**b**) system interface of the flicker-based animated SPLOM [127].

5.3. Flow and Stream

The analysis and mapping of geographic mobility data is rather essential for comprehending the spatiotemporal dynamics of datasets. Unfortunately, visualizing large mobility data and grasping the complex patterns embedded therein are still challenging research issues because of limited map space and numerous connections. This section divides origin–destination streaming data into spatial view-based flow visualization and non-spatial view-based flow visualization. The main distinction is whether the nodes or locations are fixed in the view because locations are meaningful and critical for interpreting patterns.

1. Spatial View-Based Flow Visualization

A flow map is an abstract timeline that is usually applied to depict changes in event subjects between locations, such as population migrations, import and export trades, etc. Tobler [128] studied a flow map at a very early stage and plotted a US population-migration map. Figure 49a shows that the edges in his flow map suffers from severe overlapping. Doantam et al. [129] proposed an edge-routing technique and applied it to this case, tackling the poor visualization effects caused by edge crossings (see Figure 49b). However, the results for a larger case (e.g., imports to China) still cannot meet the requirements. Selassie et al. [130] employed an edge-bundling technique [113] that aggregates similar edges into a bundle by bending the edges, revealing patterns of connectivity and symmetry without sacrificing spatial intuitiveness (see Figure 49c).



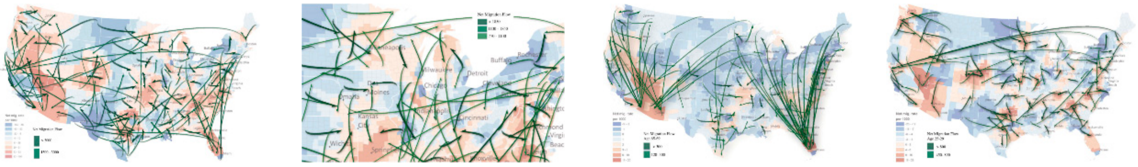
(**a**) (**b**) (**c**)

Figure 49. Flow-map visualizations at an early stage: (**a**) migration flow map of the United States Census [128]; (**b**) migration map of outflows from Colorado by using edge routing [129]; and (**c**) domestic flights in the United States by using the edge-bundling technique [130].

Common flow visualizations based on spatial views are elaborated below:

- Flow map based on flow selection

Although the above-mentioned methods effectively solve the cluttering problem in small-dataset cases, problems such as information loss, heavy dependence on user interaction, modifiable area unit and normalization still exist in the context of trajectory big data. Guo et al. [131] proposed a flow-selection method to generate a clear and wealthy flow visualization, whose basic idea is to find a subset of smooth streams that accurately represents the major flow patterns. These authors analyzed and visualized a net migration flow map of internal migration datasets from the 2000 United States Census (see Figure 50a), which smoothly supports multiresolution flow mapping (Figure 50b) and hierarchical flow mapping (Figure 50c,d).

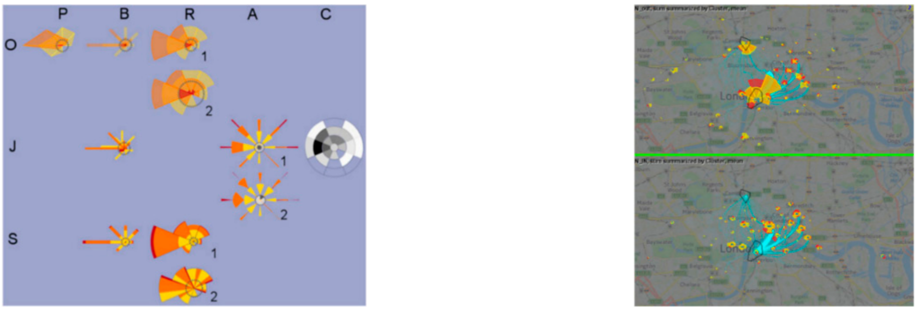


(**a**) (**b**) (**c**) (**d**)

Figure 50. Flow-selection visualization for the United States migration [131], showing: (**a**) overall net migration flows; (**b**) zoomed-in flow map; (**c**) flow map for senior migrants; and (**d**) flow map for young migrants.

- Difference diagrams

To understand the spatiotemporal patterns and long-term trends of mass-mobility behaviors, Andrienko et al. [132] developed a general procedure to analyze spatial-flow situations based on temporal dynamics, where the spatial abstraction aggregates Origin–destination (OD) flows that share a common origin or destination by the ranges of the direction and distance. This approach reduces the data dimensionality and enables a visual representation of flows via diagram maps, thereby avoiding line crossings, which are inevitable in flow maps with links denoted by lines. Figure 51a shows several variants of flow-diagram designs. The authors visualized a bicycle-trip dataset in London, indicating the flow situations of time intervals or representative flows of temporal clusters. Figure 51b shows OD flows that originated from or terminated at two selected locations with highly aggregated outflows.



(**a**) (**b**)

Figure 51. Difference diagram visualization [132]: (**a**) column labels represent shapes, and line labels represent methods for combining components; and (**b**) OD-flow subsets with incoming and outgoing locations displayed by flow lines.

- Chorisograms

A chorisogram refers to visualizing spatially interactive patterns by using statistical symbols as pattern indicators, which only focuses on interactive patterns between the regions of interest and their outside world instead of the representation of the entire interactive OD. The combination of multiple statistical patterns is interpreted as two or more interrelated spatial elements that may share a common quantitative indicator, such as a time series. To ensure pattern legibility, the combination of statistical patterns and methods thereof should not be overly complicated, enabling the quick judgment of quantitative variations and changing characteristics of each cluster via visual inspection. The strength of this approach lies in the spatial interactive information of each region can be well integrated and expressed. However, overlapping may occur in the map as the indicated regions grow, and individual symbols may have unequal sizes, making smaller patterns harder to observe. Figure 52 shows how Andrienko et al. [132] visualized car trajectories between various sites in Milan based on the chorisogram method.

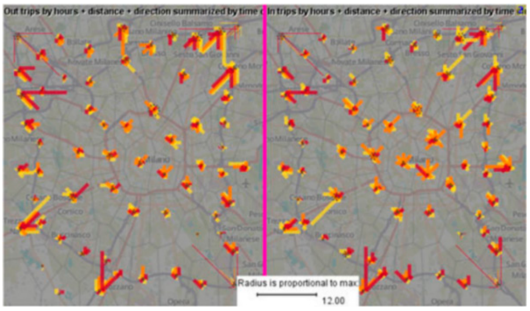

Out trips by hours • distance • direction summarized by time
In trips by hours • distance • direction summarized by time
Cinisello Balsamo
Bollate
Novate Milanese
Sesto San Giovanni
Milano
Corsico
Buccinasco
San Donato Milanese
Opera
Radius is proportional to max
12.00

Figure 52. Traffic-flow visualization of driving data in Milan, with the spatial patterns between sites [132].

2. Non-Spatial View-Based Flow Visualization

To avoid visualizing flows with crossing lines, many non-spatial view-based flow-visualization studies have been performed, focusing on ordered matrices [133], interactive OD maps [134], and exploratory visualizations [134,135]. We provide several examples for elaboration:

- OD patterns

OD patterns summarize people or vehicles' movements across geographical regions [136], and the analysis of these patterns facilitates the comprehension of urban dynamics and human activities. For example, Zeng et al. [137] proposed waypoints-constrained OD visual analytics to help users analyze OD patterns related to trajectories that successively pass through specific links or waypoints in the transportation network (see Figure 53a). Automatic detection of OD clusters helps reveal traffic surroundings and social functions. Figure 53b is a visual analytics system designed by Lu et al. [138], which integrates an interactive trajectory filter and visual OD-pattern exploration to explore OD patterns in urban areas of interest based on taxi trajectories. The authors provided a suite of graphical filtering tools to select trajectories related to a certain region of interest, see their demo video at http://vimeo.com/45078794;

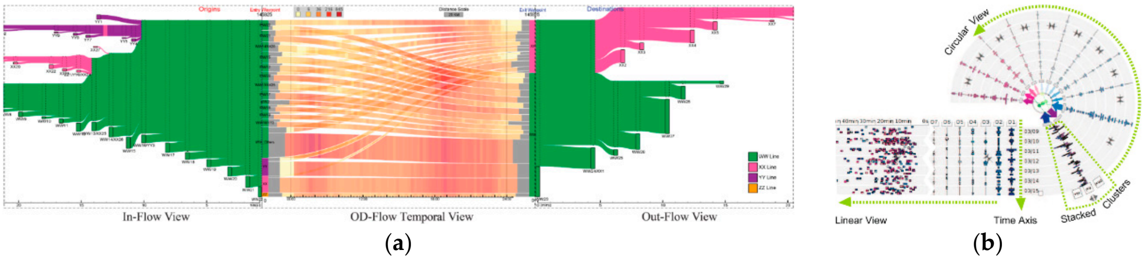
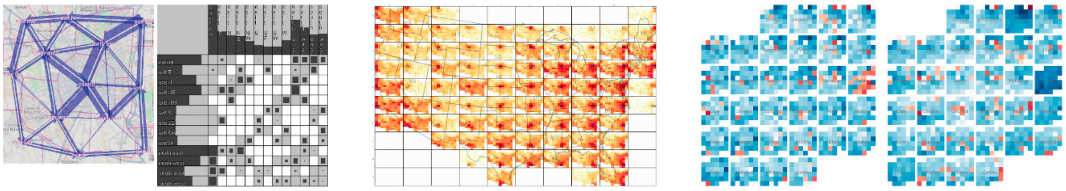


Figure 53. Two examples of OD visual analytics: (**a**) waypoints-constrained OD map with in-flow view (left), OD-flow temporal view (middle), and out-flow view (right) [137]; and (**b**) adapted OD-wheel design [138].

- OD matrix

OD matrices are derived from the matrix representation of a map [137]. Andrienko et al. [139] summarized a flow map into an OD matrix when analyzing trajectory data in Milan (see Figure 54a). In a classic OD matrix [140], origins and destinations are indicated as the rows and columns, where the cells are colored to denote the magnitude of the corresponding flow or connection. However, the geographic aspect is ignored. Wood et al. [134] proposed an idea of embedding to solve the spatial-occlusion problem of OD matrices (see Figure 54b). When studying population migration data for the United States, the authors nested the same two-dimensional matrices into each element cell in the divided regions of the two-dimensional matrix. Thus, a B-area cell nested within an A-area cell denotes flow from A to B. Initially, the OD matrix was only applicable to spatial regions divided by regular grids because visualizing spatial-migration patterns with respect to immigration issues is difficult. To depict population migration between different districts of Ireland, Slingsby et al. [141] manually arranged these regions into matrices, more effectively depicting spatial patterns of county-to-county migration and applying the OD matrix to spatial regions as divided by irregular grids. The authors visualized migration plots for years of 1851 and 1911 to compare the migration difference (see Figure 54c).



(**a**) (**b**) (**c**)

Figure 54. OD matrix visualizations: (**a**) traffic flow map (left) and corresponding OD matrix (right) in Milan [139]; (**b**) embedding-based population migration map of the United States [134]; and (**c**) percentage difference in spatial migration [141].

- Flow maps

For non-directional statistics without trajectory information, users may have trouble exploring events and interactions in space and time. Kim et al. [142] developed a novel flow-analysis system for extracting, representing, and analyzing flow maps of non-directional spatiotemporal data without trajectory information. The authors estimated a continuous distribution of these events in space and time and extracted flow fields for spatiotemporal changes via a gravity model, visualizing the spatiotemporal patterns in the data. Figure 55 shows their collection and analysis of Twitter messages based on the 2015 Super Bowl event, investigating message flows in a certain area of the system.

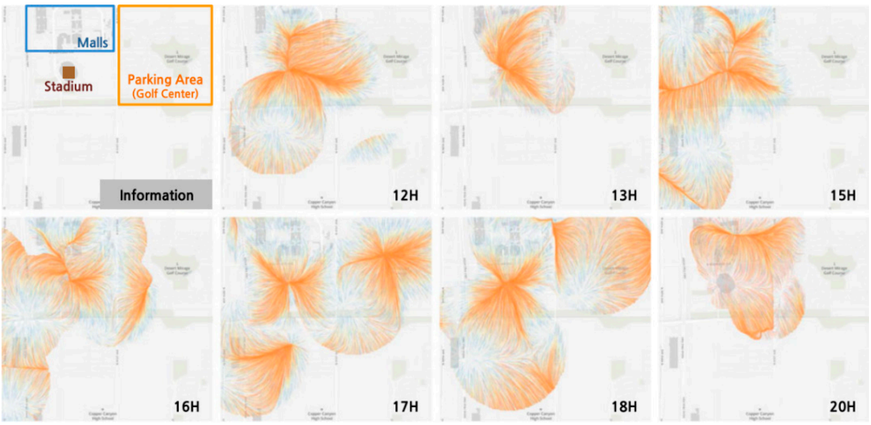


Figure 55. Flow maps detecting converging flow patterns [142].

6. Comparative Analysis of Visualization Techniques

6.1. Universal Multivariate Visualization Techniques

How do we better present trajectory events to enhance and magnify reasoning? Many solutions for visualizing multivariate and multidimensional trajectory data have been studied and analyzed in the literature. As the basic variable-analytics methods of trajectory visualization, icons, semantics, and word clouds can be combined with any dimension-targeted visualization techniques to frame novel visual-representation approaches. The icon method produces a compact form to encode various variables or attributes distinctly and has extensive solutions, generally colorful and intuitive visualization results, exhibiting excellent performance when visualizing particular-dimensional attributes, especially direction information. This method is therefore widely applied in trajectory visualizations [143,144] but could exhibit inadequate adaptability. Designing an effective icon is never easy, even though most existing designs have gone through a lasting process of evolution, modification, and normalization. One of the most common criticisms of icon methods is that implicit deviations exist in most mappings, i.e., some graphical features or relationships between features are more perceptible than others. Moreover, visual encodings with adequately high resolution are of limited number. This technique is applied to low-dimensional datasets to avoid over-plotting.

In trajectory data processing, we may be concerned with the rich information features represented by the trajectory data, such as the scenario involved, operation mode, etc., but many trajectory data themselves do not include such information features. Moreover, many visualization studies have failed to regard the impact of location data sparsity on prediction accuracy. In this regard, research on user-location prediction typically mitigates data sparsity issues of individual users by combining mobility data for socially related users, introducing traffic condition data, or utilizing multi-source mobility data. These can be enabled by techniques of semantics and word cloud, which artificially assign context features or behavioral-pattern features to these trajectory data. Therefore, these approaches are better adapted to the real environment and mining deep-level trajectory information. Methods of semantics and word clouds could require combining other dimension-targeted visualization techniques to address specific issues, in which most hidden semantic behaviors can be discovered. The trajectory information-category analysis in such cases is effortless and effective. It is worth mentioning that, although word clouds seem more attractive than other techniques as a popular visual representation, words of the same size are hard to compare since human eyes are insensitive to the magnitudes of areas or words. A longer word occupies more display space; thus, words involving different weights could be mistakenly observed to be the same size. We list the strengths and limitations of these three techniques in identifying different dimensional data instances in Table 1.

Table 1. Overview comparing strengths and weaknesses of multivariate visualization techniques.

	Icons	**Semantics**	**Word Clouds**

Strengths	Intuitive;
	Special-dimensional attributes;
	Rich configuration
		Detailed display;
		No occlusions;
		Legibility
			Strong highlighting;
			Stylish and novel;
			Enrichment

Weaknesses	High occlusion probability;
	Inadequate adaptability;
	Implicit deviation
		Multiple implementations;
		Memory dependency;
		Combined implementations
			Low resolution;
			More display space occupied;
			Inherent deviation

6.2. Low-Dimensional Data-Targeted Visualization Techniques

We have covered visualization methods that target low-dimensional data. Of course, these techniques are more but not just applicable for low-dimensional data; sometimes they might function as fine approaches of information integration and visualization for a minor class of high-dimensional data, of which the final results indeed fit most people's cognition and comprehension. Here, we take stacking and density-map methods as exemplifications.

Although the density map is more suitable for low-dimensional trajectory data, there is surely the possibility of high-dimensional trajectory data application. We derived a 4D time density by modeling the movement path and velocity. Compared to the 3D space–time density, the density-calculation algorithm was modified to combine the fourth-dimensional time density into a 3D geographical space instead of a 2D space. Figure 56 shows the application of a 4D time-density trajectory on a real dataset, which successfully captured the temporal density variation of a large trajectory dataset and visually identified the specific spatiotemporal patterns of aircraft movement.

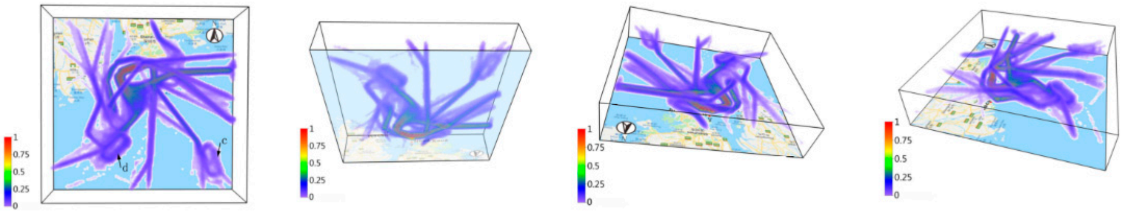


Figure 56. 4D time density of flight routes displayed in different directions [145].

Stacking approach can also involve high-dimensional data visualization, but not common. As an example, we sought to propose an elevation model for multiattribute trajectory sets along the idea of stacking and applied this model to the behavioral description of a mining truck in a mining area in Inner Mongolia, China. Figure 57 shows the analysis result of the truck that maps and renders the elevation stacking model of the trajectory, extending the 3D representation of 2D trajectory stacks into a 4D-like representation of 3D trajectory stacks, i.e. a high-dimensional trajectory-data visualization.

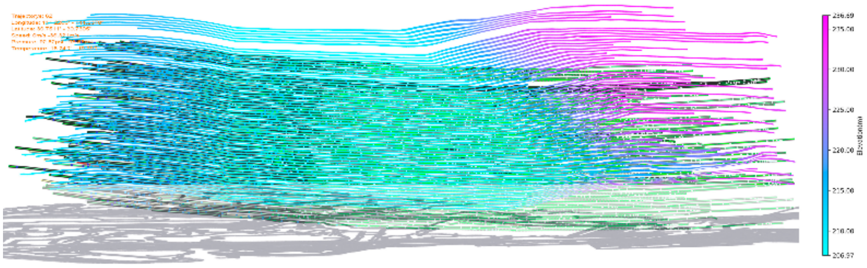


Figure 57. Elevation model with elevation ribbons superimposed on an attribute layer.

How to make better decisions from these visualization techniques when analyzing multisource trajectory data? In general, although space–time cubes effectively match temporal information, their availability is often limited by 3D cognitive capabilities of human beings [146]. Therefore, this method is typically used for the visualization of simple paths over relatively short periods instead of complex, long-span trajectories. Nonetheless, from the perspective of effectiveness, STCs are often used in combination with interactive filtering and clustering and applied to events or trajectories for clutter reduction and aggregation [26]. Fundamentally, trajectory stacking belongs to an enhanced STC model, yet its stacking method has new extensions for the thematic attributes of the trajectories. In such a stacking mode, the vertical dimension denotes temporal attributes, the horizontal dimensions indicate geospatial positions, and a diagonal represents certain spatiotemporal attributes. If necessary, we can also select other trajectory attributes to sequence trajectories on the z-axis, which effectively supports the cognitive and exploratory spatiotemporal analysis of trajectory attribute data. However, as an intuitive multisequence data-visualization method, stacking graphs are applied prudently in visual analytics because of defects such as weak comparisons, poor scalability when the sequential-data amount grows, etc. Further, this solution only works well for hard- and soft- constrained trajectories with similar geometries; once the tracked movements lose this constraint or even become chaotic, this approach no longer applies.

Density maps effectively solve the problem of visual scalability; these maps can be used in detecting clusters from emerging spatial big data and are adapted to the spatial division of most trajectories. This approach is especially useful when single trajectories are of no interest. The concept of a density map can be extended to multiple density fields that represent both densities and other attributes, enhancing the expressivity and flexibility of density maps. Multiple density images that are generated with different parameters or represent different attributes can be combined into a single composite density map using special operators [42]. Examples show that such maps can effectively differentiate moving and anchored objects or highlight anomalous movements. Similar to the density map, heatmaps reduce the data-rendering workload by data aggregation, which can effectively mitigate overlapping caused by massive trajectories. However, this advantage is more reflected in the analysis of macro trajectory data, supporting the conditional spatial division of most trajectory data. Obviously, heatmaps do not retain any information regarding a particular trajectory, so comparison between trajectories is impossible. Additionally, some individual points could face the over-binning problem.

In contrast, meshing approaches are more of diverse complexity to some extent, which may well identify structural or topological features of cities but involve no real traffic information and thus lack data persuasiveness. We generally believe that this form of exploration environment enforces certain constraint on related visualization of attribute features and effects the accuracy of data-analysis algorithms. Time-series visualization can depict a single dimension of trajectory data in a simple and direct manner with hierarchical granularities. However, high variability typically exists between trajectories. This approach can hardly capture the evolution of complex trajectory events; even regular cases could involve some anomalies.

It cannot be ignored that there are very few studies on the movement patterns of animal collectives, which helps analysts find answers to specific questions regarding the movement behavior of an animal collective. Transformation approach can well discover hidden knowledge of animal-trajectory behaviors, which significantly facilitates the direct comparisons between research sites and species. Moreover, with the transformed trajectories, we can study and compare the movement behaviors of individuals within the group. We compare the strengths and weaknesses of Low-dimensional visualizations in Table 2.

Table 2. Overview comparing strengths and weaknesses of low-dimensional visualization techniques.

	Space–time Cube	**Stacking**	**Density Map**	**Heatmap**	**Meshing**	**Time Series**	**Transformation**
Strengths	Match the temporal information; 3D rendering; Query and integration	Attribute extension; 3D rendering; Avoid cluttering	Broad visual space; Cluster detection; Avoid overlapping	Broad visual space; Cluster detection; Avoid overlapping	Structural features; Privacy protection; Highlighted correlation	Intuitive results; Quick interpretation; Striking contrast	Trajectory-set behaviors; Individual trajectory behaviors; Behavior comparison
Weaknesses	Ignored branch time; Spatial limitation; Combined implementation	Comparative issue; Poor temporal scalability; Object constraint	Separate rendering; Comparative issue; Cannot interpret the exact values	Separate rendering; Comparative issue; Over-binning	Complex algorithm; Poor data persuasiveness; Attribute constraint	More display space occupied; Strict data limitation; Comparative issue	Combined implementation; Poor scalability; Insufficient information

6.3. High-Dimensional Data-Targeted Visualization Techniques

High-dimensional trajectory data can be considered analogous to low-dimensional trajectory data. In low-dimensional space, we typically employ Euclidean distance as the similarity measure between data, but the concept of similarity no longer exists in many cases of high-dimensional space, which poses a severe challenge to the visualization of high-dimensional data. How should this be operated? We have already discussed the practical operation of low-dimensional visualization and concluded that certain operations produce visualizations that may be more appropriate for certain tasks. This type of operation, which is not absolute, can benefit from useful recommendation based on more general knowledge about visual perception and common sense. Regarding data types, high-dimensional visualization can inform the applicability of certain operations in many cases. Thanks to the evolvement of machine learning, the dimension-reduction methods serve well and develop rapidly. Facing the deficiency of reduction-algorithm efficiency, the algorithm performance can be optimized by devising more efficient index structure and using incremental and parallel algorithms, and even invalid issues can be redefined and revived. However, the problems brought by dimensionality reduction are also obvious. The intention of dimensionality reduction is to filter out redundant and meaningless information in original dimensions, but this process may inevitably lose useful information. To avoid this drawback, another approach decides to directly present the pairwise relationship of dimensional association using scatter plot matrix, instead of reducing the dimensionality. In terms of scatter plots, this approach exhibits positive visualization effects on data of any dimensionality, thus is classified in Section 5. However, the growing dimension has quickly caused users headaches: assuming a 10-dimensional scatter plot matrix, totally 100 small scatter plots could be a time- and brain-consuming task. Therefore, we expect another method without both dimensionality reduction and large quantities of plots for users to interpret, hopefully only single plot involved. Typical and representative techniques of this approach include parallel coordinates, RadViz, star coordinates, etc. We elaborate these methods below.

The dimensionality reduction of data can simplify high-dimensional data, dramatically reduce the complexity of clustering algorithms, and partially display abstract and imperceptible high-dimensional data structures in low-dimensional space. However, this process tends to lose data information, so that low-dimensional space cannot accurately reflect the original high-dimensional data relationship. The higher the dimension, the heavier the information loss becomes. Long-distance data separated in high-dimensional space could be mistakenly recognized into a same cluster after dimensionality reduction, and vice versa. Many existing dimensionality-reduction methods are based on the optimization of data's statistical features. These approaches can minimize information loss but cannot provide clear semantic information regarding the dimensionality-reduction results, which increase the difficulty of data interpretation. Although the users might comprehend context knowledge or understand the semantic connotation of the data, this information is not considered by most dimensionality-reduction methods; thus, the reduction results may diverge from the user's perception, thereby reducing its utility. The current data dimensionality-reduction methods still have problems regarding poor semantics and mismatching user backgrounds, which poses difficulties for data analytics and application in various aspects of data modeling, theoretical construction and interpretation, etc. One solution is not to display the entire dataset but to visualize only its subsets. Analogous to dimensionality reduction, projection method reduces the dimensionality of the dataset by preserving the distance as much as possible, aiming to find informative projections and transformations for multivariable trajectory data to visually encode them with a spatial-location channel.

Hierarchy-visualization techniques are beneficial for addressing the inconsistency of hierarchical data structures, and most research has focused on the challenge of displaying large hierarchical trajectory sets in a comprehensible manner [147]. The strength of this approach is its applicability to continuous trajectory datasets. However, once the dimensionality exceeds a certain degree, the outcome is susceptible to visual clutter, and the relationship between the coordinates of each hierarchy turns opaque. As a representative, treemap's spatial efficiency saves the space of hierarchical data and enables comparisons between the depth of each node and the correlation between the nodes, discovering the spatiotemporal patterns and anomalies in the data. Although this approach effectively uncovers relationships between various dimensions, it fails to consider the relationship between hierarchical structures and attributes. Fortunately, pixel method well tackles this problem, showing excellent performance in visualizing trajectory big datasets with a certain dimensionality. Numerous applications that mine correlations between multidimensional attributes have employed this technique. If analysts might further discover trajectories of interest between origin and destination area, columnar pixmaps enable the comparison and extraction with regard to trajectory events of interest. Pixel-based techniques exhibit relatively better scalability in multidimensional trajectory-data visualizations.

However, human performance can be affected by many factors, such as screen resolution, working memory and attention requirements. Meanwhile, this type of visualization suffers heavy costs in online interactions and animated transitions because of the high data-rendering density.

One of the major advantages of applying parallel coordinates in trajectory-dataset analysis is their excellent scalability and data dimensionality: as the trajectory-data attribute categories increase, all we need to do is add more axes, which work fine up to limitations that are imposed by the display space. However, parallel-coordinate systems actually have an upper dimensionality limit because of the limited visual range and resolution of human eyes. These systems are suitable for datasets with small data volume but high dimensionality, having fine performance in outlier mining. When the dataset involves hyperdimensional data, a huge display space is required for full representation, hence a severe problem of line overlapping, in which users can hardly interpret the correlation between attributes of each dimension. This problem can be solved by dividing the high-dimensional data space into smaller subsets. Another strength of this technique is the applicability for representing the correlation between dimensions and analyzing the interdependence between variables. Specifically, parallel lines between two axes indicate a positive correlation, whereas intersecting lines denote a negative one. Although parallel coordinates facilitate the discovery of differences between interactive patterns and related forms of multivariate relationships between regions, this approach cannot reflect spatial patterns because only one indicator of the interactive data is expressed. A multivariate relationship involves multiple pairs of parallel coordinate axes, and curved lines (instead of segmented lines) can improve the traceability [148]. Unfortunately, replacing lines with curves could destroy some important geometric features of the parallel coordinates. In brief, curved parallel coordinates fail to explicitly highlight multivariate relationships. Furthermore, parallel coordinates result in asymmetric visual representations of positive and negative correlations, thus restricting the ability to display multivariate correlations: since the lines form crossing patterns between the axes, a negative correlation tends to be detected, while discovering a positive correlation is more difficult. Positively correlated samples are observed as segments that intersect outside their own pair of axes; or if the slope of the linear regression line approaches one, these samples are observed as practically parallel segments. These patterns become hard to observe, resulting in a general underestimation of the correlation [149]. The technique of parallel-coordinate visualization has also evolved into RadViz [150], circular parallel coordinates [151], hierarchical parallel coordinates, star coordinates [152], etc., which remedy shortcomings in parallel-coordinate systems and further improve the quality of visualization. However, these techniques exhibit ordinary performance in big data-visualization applications because of the fundamentals of representing multidimensional space with multiple axes. Here, we focus on analyzing the visualization technique of star coordinates.

The star-coordinate technique enables flexible, user-driven mappings of multidimensional trajectory data into a two-dimensional space [97,152]. Meanwhile, this approach effectively reveals high-dimensional connections between variables and data points in multivariate analysis. A strength of this layout is its efficiency in terms of displaying periodic patterns, while a disadvantage is its low spatial efficiency. When merging multiple attributes of trajectory data by weight, star coordinates can reflect the statistics of high-dimensional data and show whether the distribution of high-dimensional data is balanced. Compared to other multidimensional visualization methods (e.g., scatter plot matrices and parallel coordinates), the star-coordinate model is probably the most scalable technique for visualizing large datasets. Here, "scalability" refers to data processing and visual representation. We should be aware of three points: (1) Although various underlying mapping models exist for star coordinates, such as the Kandogan model [152] and VISTA model [101], the visual design of star coordinates has its unique characteristics. This design extends the traditional two- or three-dimensional coordinate systems to K-dimensional coordinate systems, which can obviously present more dimensions than scatter plot matrices and parallel coordinates. In fact, both methods are generally used to explore data with less than 10 dimensions, while star coordinates can handle tens of dimensions [153]. In particular, star coordinates can perform better cluster visualization with multiple dimensions. Conversely, only two dimensions can be observed in scatter plots, and clusters may suffer severe overlapping in parallel coordinates. (2) By means of a density-based representation, star-coordinate visualization can be extended to many points. Thanks to the density-preserving properties of the underlying mapping model, points can be aggregated and density information can be displayed, which is critical when the number of records grows huge (say, tens of millions to billions of records). (3) Instead of computing the pairwise distances between records, which requires unacceptable costs, the clustering-visualization method based on star coordinates utilizes the attributes of the underlying mapping model to partially preserve distance connections, which is of great significance for processing big data. In summary, star-coordinate plots have two key advantages in terms of reflecting data distribution. First, these plots are an easily realizable approach, whose simplicity enables extension into more complex data-visualization tasks. Second, these plots only require one constant time to render each data point, providing a fine characteristic of linear time complexity with respect to data size, which makes these plots an ideal choice for real-time, data-intensive visualization applications.

As another radial-visualization approach, RadViz not only reflects the changing patterns of each trajectory attribute, but also denotes conditions of each type of feature attributes in the overall trajectory set. However, when choosing between these methods, one should consider whether the inherent nonlinear normalization of RadViz is advantageous for the trajectory-dataset analysis. Radar charts are primarily applicable in highlighting outliers and commonality, or mainly used for ordinal measurements when each variable in one chart is greater than that in another, which exhibits interactive and auxiliary-classification features in data-mining classification. However, radar charts have been criticized as poorly suited for trade-off decisions. The overall comparison is provided in Table 3.

Table 3. Overview comparing strengths and weaknesses of high-dimensional visualization techniques.

	Dimensionality Reduction	**Projection**	**Hierarchy**	**Pixmaps**	**Parallel Coordinates**	**Radial Graph**

Strengths	Algorithms of simplicity and diversity;
	Excellent perceptible effect;
	Revivable
		Intuitive;
		No occlusions;
		Good rendering performance
			Continuity;
			Spatial efficiency;
			Anomaly detection
				Correlation mining;
				Comparison and extraction;
				Healthy scalability
					Flexible configuration;
					Scalability;
					Unlimited variables
						Periodic;
						Distributional-balance detection;
						Scalability

Weaknesses	Poor semantics;
	Weakened user contexts;
	Information loss
		Combined implementation;
		Distortion;
		Discrete
			Cluttering risks;
			Obscure to understand;
			Strict data limitation
				Dense visual display;
				Over-plotting risks;
				Subject to resolution
					Restricted correlation display;
					No spatial patterns involved;
					Upper dimensionality limit
						Low spatial efficiency;
						Poor decision making;
						Combined implementation

6.4. Universal Multidimensional Visualization Techniques

The visualization issue of depicting trajectories includes summarizing real-life problems encountered by users, abstracting the operations of corresponding data types, designing methods of encoding and interaction, and implementing algorithms and interactions. As universal data-state models, these visualization techniques and methods are either used to illustrate the above-mentioned issues, or to support different visual views—the operations themselves are of diversity. We have covered that, although the heatmap method solves the overlapping problem well, this approach can only be used as a qualitative analysis method, failing to meet the requirements of quantitative analysis or indicate geographical positions accurately enough. Clustering graph is an excellent visualization method for analyzing massive trajectory data, which facilitates statistics and displays of global information, as well as prevents data loss. Besides, clustering can well process unevenly distributed data.

In contrast, scatter plots provide powerful selecting and filtering functions for data exploration. Compared to heatmaps and density maps, this approach can also show the data variations in different areas within a certain reach of the trajectory big data but without a progressive process. Scatter plots serve well for visualizing a limited number of relatively meaningful dimensions but are generally not suitable for displaying all dimensions simultaneously because of visual cluttering or over-plotting. As for bubble charts, this technique only functions in applications where the third dimension is not required to be accurately identified, because users are not good at judging the size of an area. Rather, as an effective tool for recognizing correlations, clusters, outliers, and other interests, scatter plot matrix works well for high-dimensional trajectory data. The motivations behind this choice are that: (1) cognizing data correlation in low-projection views is generally more intuitive; and (2) this method experiences less visual cluttering and over-plotting compared to parallel coordinate visualization. Generally, a strength of this method is that it is not subject to the dataset's size. Furthermore, since information is derived from comparisons between pairs, all the elements do not need to be displayed on the screen simultaneously, which considerably reduces the visualization difficulty caused by increased dimensionality.

In flow visualizations, the spatial flow patterns specify the location of the origins, while graphic rendering methods in non-spatial domains are generally not directly applicable to flow mapping. The strength of spatial view-based flow visualization is that the plotted "timeline" not only indicates temporal and spatial characteristics, but also enables the representation of quantitative thematic attribute information, making a thematic map-visualization technique that adequately expresses event processes. However, only univariate thematic attributes whose quantities change over time are supported. Cross-line clutters can also obscure spatial patterns, while arbitrary intersections may imply non-existent patterns and neglect different densities or scales of flow clusters. These problems can be solved by non-spatial view-based flow visualizations. These methods are based on spatial transformations and can minimize the distortions of their relative spatial positions with respect to each other, to some extent avoiding visual cluttering without significant deviations. However, spatial distortion might complicate perception because the overall spatial pattern of flows is broken into multiple location-specific patterns, thus failing to highlight the spatial-flow patterns. Table 4 summarizes the pros and cons of universal multidimensional visualization methods.

Table 4. Overview comparing strengths and weaknesses of multidimensional visualization techniques.

	Clustering	**Scatter Plot**	**Spatial View-based Flow Visualization**	**Non-spatial View-based Flow Visualization**

Strengths	Quantitative analysis;
	Massive data supported;
	Global statistics and display
		Selecting behaviors;
		Filtering behaviors;
		Cluster detection
			Clear spatiotemporal features;
			Intuitive results;
			Quick interpretation
				Avoid occlusion;
				No deviation;
				Privacy protection

Weaknesses	Combined implementation;
	Restricted attributes;
	Individual rendering
		Non-progressive process;
		Visual cluttering;
		Individual rendering
			More display space occupied;
			Visual burden;
			Comparative issue
				Complicated perception;
				Poor overview;
				Combined rendering

7. General Discussions

As the complexity of multivariable trajectory data has grown and the abstraction of data features has increased in recent years, the limitations and uncertainties of related visualization techniques have gradually emerged. Urgent issues include choosing appropriate visualization techniques for different data categories and derived features, better coordinating multiple data attributes and their display results, reducing informational clutter and visual occlusions caused by data multidimensionality, establishing a universally applicable evaluation framework, and visualizing data in an apt manner for utilization, navigation and comprehension. Meanwhile, unprecedented opportunities and challenges have arisen: on the one hand, we have reason to believe that more trajectory data resources will become accessible and analyzable; on the other hand, the scalability, heterogeneity, sparsity and uncertainty exhibited in such data will become more significant, requiring further cognitive experiments for the verification of which visualization method better meets user cognition. Here, we focus on two major and challenging issues that have existed or are forthcoming.

1. Data issues. The basis on which the big-data visualization relies is data. Trajectory data have numerous sources derived from heterogeneous environments; the integrity, consistency, and accuracy of the data sources in this case are hard to guarantee. Although preprocessing addresses numerous data-quality problems, uncertainties still exist. Massive trajectory data could potentially expose private information such as behavioral characteristics, hobbies, and social relationships that are generally concerned with fundamental interests of the users. Moreover, clarifying sampling errors or ambiguity due to increased focus on privacy protection or producing an appropriate visual design unsusceptible to data-quality issues have become particularly importance. With the advancing data-acquisition technology, explosively increasing data dimensionalities and emerging high-dimensional data types, sometimes data analysis with existing visualization methods can suffer heavier detail loss, or even cannot be directly performed. Additionally, most people can hardly perceive and comprehend spaces above four dimensions because of human-brain limitations, thus maximizing the amount of details becomes more difficult. Therefore, we should design a big-data visual analytics system whose perceptual scalability and interactive scalability depends on the visualization accuracy and computer processing power rather than the data scale.
2. Analysis issues. Big-data visual analytics is a process of human–machine interaction. However, there is a vacancy for recognized scientific evaluation mechanism to design and evaluate visual representations for matching mental images, because multifaceted cognitive divergence has existed between users and experts in the application field. Therefore, the multilevel and multi-granular task distribution of human–machine interactions and fulfillment of user-centered system frameworks should be further studied and verified, since ordinary individuals in any field will require trajectory analysis in the big-data era of the future. In current visualization systems, data analysis is essentially limited to descriptive or exploratory analysis, while practical problems require answers with clarity, predictability, and causality. Current visual analytics typically only support a single data type, which does not facilitate the investigation of implicit relationships between multiple data sources. Furthermore, the increasing complexity of analytical tasks requires synthesizing heterogeneous types of information in trajectory big-data analysis [154]. It turns out that visualizing all aspects of trajectory data in a single static view is unacceptable; such attempts often trigger information overload and visual clutter. Conversely, multiple coordinated views require explorations back and forth between these views when simultaneously analyzing spatiotemporal data and other attributes of trajectory data, because of the views' multiple foci. Integrating multiple attributes into one view to reveal the interlinkages between attributes and avoid over-plotting is a challenge for trajectory-data visualization.

8. Conclusions

Visualization plays a crucial role in the field of multiattribute spatiotemporal trajectory-data analysis and cognition [155], exhibiting features of Intuitiveness, interactivity, dynamicity, multidimensionality and integration. This paper provided brief introductions of representative applications of various visualization methods and techniques for trajectory data and their latest developments. The choice of trajectory visualization method is flexible and diverse: one may use the same visualization technique in different cases or different techniques in one case to collaboratively display related spatial and temporal patterns. Although conventional visualization methods still have extensive applications, the advent of the big-data era has motivated the emergence of more powerful trajectory-visualization techniques, involving modifications of previous methods, combinations into novel approaches, or the utilization of technologies from other visualization domains to trajectory datasets. In other words, trajectory visualization is required to extract fundamental elements of spatiotemporal information and thematic attributes from geographical phenomenon, revealing the spatiotemporal context and attribute features of trajectory events for users to grasp, and reflect behavioral changes in the information over the temporal and spatial domains in either a static or dynamic manner of interaction. With the rapid development of the Internet and continuous advancement of artificial-intelligence technology, research on trajectory-data visualization still has unlimited potential and possibilities; we also expect more efficient techniques and methods to emerge.

Author Contributions: All authors contributed to this review paper. J.H. had the biggest role of initiating the review, taking the lead on identifying relevant literature, writing and organization, and coordinating input from other authors. Y.C. and X.T. provided input on overall organization of the paper; H.C. identified selected research to include in the review, and contributed to writing and editing the text. Y.Z. collected the uncertain data in the process of writing. All authors revised the manuscript.

Funding: This research was funded by the Fundamental Research Funds for the Central Universities grant number 2010YD06.

Acknowledgments: This material is based on work supported by the Fundamental Research Funds for the Central Universities under Grant No. 2010YD06. This work was performed at and supported by the College of Geoscience and Surveying Engineering, China University of Mining and Technology, Beijing.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Keim, D.; Andrienko, G.; Fekete, J.D. Visual analytics: Definition, process, and challenges. In *Information Visualization*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 154–175.
2. Cao, N.; Lin, Y.; Sun, X.; Lazer, D.; Liu, S.; Qu, H. Whisper: Tracing the spatiotemporal process of information diffusion in real time. *IEEE Trans. Vis. Comput. Graph.* **2012**, *18*, 2649–2658. [PubMed]
3. Buschmann, S.; Trapp, M.; Döllner, J. Animated visualization of spatial–temporal trajectory data for air-traffic analysis. *Vis. Comput.* **2016**, *32*, 371–381. [CrossRef]
4. Möller, T.; Haines, E.; Hoffman, N. *Real-Time Rendering*, 3rd ed.; A.K. Peters: Natick, MA, USA, 2008; p. xviii. 1027p.
5. Sheng, F. The Visual Analysis of Traffic Data Based on Semantic Extraction. Master's Thesis, Zhejiang University of Technology, Hangzhou, China, 2017.
6. Spaccapietra, S.; Parent, C.; Damiani, M.L.; de Macedo, J.A.; Porto, F.; Vangenot, C. A conceptual view on trajectories. *Data Knowl. Eng.* **2008**, *65*, 126–146. [CrossRef]
7. Zhong, C.; Zaki, C.; Tourre, V.; Moreau, G. Event-based semantic visualization of trajectory data in urban city with a space-time cube. In *Proceedings of the 3rd WSEAS International Conference on Visualization, Imaging and Simulation*; World Scientific and Engineering Academy and Society (WSEAS): Faro, Portugal, 2010; pp. 99–105.
8. Ratcliffe, J.H.; Chainey, S. *Gis and Crime Mapping*; John Wiley & Sons Ltd (10.1111): Malden, MA, USA, 2005.
9. Mburu, L.; Helbich, M. Evaluating the accuracy and effectiveness of criminal geographic profiling methods: The case of dandora, kenya. *Prof. Geogr.* **2015**, *67*, 110–120. [CrossRef]
10. Liao, Z.-F.; Li, Y.; Peng, Y.; Zhao, Y.; Zhou, F.-F.; Liao, Z.-N.; Dudley, S.; Ghavami, M. A semantic-enhanced trajectory visual analytics for digital forensic. *J. Vis.* **2015**, *18*, 173–184. [CrossRef]
11. Le, T.M.V.; Lauw, H.W. Semvis: Semantic visualization for interactive topical analysis. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*; ACM: Singapore, 2017; pp. 2487–2490.
12. Al-Dohuki, S.; Wu, Y.; Kamw, F.; Yang, J.; Li, X.; Zhao, Y.; Ye, X.; Chen, W.; Ma, C.; Wang, F. Semantictraj: A new approach to interacting with massive taxi trajectories. *IEEE Trans. Vis. Comput. Graph.* **2017**, *23*, 11–20. [CrossRef] [PubMed]
13. Bogorny, V.; Avancini, H.; de Paula, B.C.; Kuplich, C.R.; Alvares, L.O. Weka-stpm: A software architecture and prototype for semantic trajectory data mining and visualization. *Trans. GIS* **2011**, *15*, 227–248. [CrossRef]
14. Chen, S.; Yuan, X.; Wang, Z.; Guo, C.; Liang, J.; Wang, Z.; Zhang, X.L.; Zhang, J. Interactive visual discovering of movement patterns from sparsely sampled geo-tagged social media data. *IEEE Trans. Vis. Comput. Graph.* **2016**, *22*, 270–279. [CrossRef]

15. Seifert, C.; Kump, B.; Kienreich, W.; Granitzer, G.; Granitzer, M. On the beauty and usability of tag clouds. In Proceedings of the 2008 12th International Conference Information Visualisation, London, UK, 9–11 July 2008; pp. 17–25.
16. Cui, W.; Wu, Y.; Liu, S.; Wei, F.; Zhou, M.X.; Qu, H. Context-preserving, dynamic word cloud visualization. *IEEE Comput. Graph. Appl.* **2010**, *30*, 42–53. [CrossRef]
17. Leginus, M.; Zhai, C.; Dolog, P. Personalized generation of word clouds from tweets. *J. Assoc. Inf. Sci. Technol.* **2016**, *67*, 1021–1032. [CrossRef]
18. Ertl, T.; Chae, J.; Maciejewski, R.; Bosch, H.; Thom, D.; Jang, Y.; Ebert, D.S. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In Proceedings of the 2012 IEEE Conference on Visual Analytics Science and Technology (VAST), Seattle, WA, USA, 14–19 October 2012; pp. 143–152.
19. MacEachren, A.M.; Jaiswal, A.; Robinson, A.C.; Pezanowski, S.; Savelyev, A.; Mitra, P.; Zhang, X.; Blanford, J. Senseplace2: Geotwitter analytics support for situational awareness. In Proceedings of the 2011 IEEE Conference on Visual Analytics Science and Technology (VAST), Providence, RI, USA, 23–28 October 2011; pp. 181–190.
20. Thom, D.; Bosch, H.; Koch, S.; Wörner, M.; Ertl, T. Spatiotemporal anomaly detection through visual analysis of geolocated twitter messages. In Proceedings of the 2012 IEEE Pacific Visualization Symposium, Songdo, Korea, 28 February–2 March 2012; pp. 41–48.
21. Bosch, H.; Thom, D.; Heimerl, F.; Püttmann, E.; Koch, S.; Krüger, R.; Wörner, M.; Ertl, T. Scatterblogs2: Real-time monitoring of microblog messages through user-guided filtering. *IEEE Trans. Vis. Comput. Graph.* **2013**, *19*, 2022–2031. [CrossRef] [PubMed]
22. Chu, D.; Sheets, D.A.; Zhao, Y.; Wu, Y.; Yang, J.; Zheng, M.; Chen, G. Visualizing hidden themes of taxi movement with semantic transformation. In Proceedings of the 2014 IEEE Pacific Visualization Symposium, Yokohama, Japan, 4–7 March 2014; pp. 137–144.
23. Wang, R. The Visualization and Analysis of Traffic Data Stream Based on Topic Modeling. Master's Thesis, Master, Hangzhou Dianzi University, Hangzhou, China, 2016.
24. Itoh, M.; Yoshinaga, N.; Toyoda, M. Word-clouds in the sky: Multi-layer spatio-temporal event visualization from a geo-parsed microblog stream. In Proceedings of the 2016 20th International Conference Information Visualisation (IV), Lisbon, Portugal, 19–22 July 2016; pp. 282–289.
25. Hägerstrand, T. What about people in regional science? *Pap. Reg. Sci. Assoc.* **1970**, *24*, 6–21. [CrossRef]
26. Andrienko, G.; Andrienko, N.; Chen, W.; Maciejewski, R.; Zhao, Y. Visual analytics of mobility and transportation: State of the art and further research directions. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2232–2249. [CrossRef]
27. Bach, B.; Dragicevic, P.; Archambault, D.; Hurter, C.; Carpendale, S. A descriptive framework for temporal data visualizations based on generalized space-time cubes. *Comput. Graph. Forum* **2016**, *36*, 36–61. [CrossRef]
28. Bach, B.; Pietriga, E.; Fekete, J.-D. Visualizing dynamic networks with matrix cubes. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Toronto, ON, Canada, 26 April–1 May 2014; pp. 877–886.
29. Kapler, T.; Wright, W. Geotime information visualization. In Proceedings of the IEEE Symposium on Information Visualization, Austin, TX, USA, 10–12 October 2004; pp. 25–32.
30. Mayr, E.; Windhager, F. Once upon a spacetime: Visual storytelling in cognitive and geotemporal information spaces. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 96. [CrossRef]
31. Carpendale, M.S.T.; Cowperthwaite, D.J.; Tigges, M.; Fall, A.J.; Fracchia, F.D. Tardis: A visual exploration environment for landscape dynamics. In Proceedings of the Electronic Imaging '99, San Jose, CA, USA, 25 March 1999; p. 10.
32. Vrotsou, K.; Forsell, C.; Cooper, M. 2d and 3d representations for feature recognition in time geographical diary data. *Inf. Vis.* **2009**, *9*, 263–276. [CrossRef]
33. Forlines, C.; Wittenburg, K. Wakame: Sense making of multi-dimensional spatial-temporal data. In Proceedings of the International Conference on Advanced Visual Interfaces, Roma, Italy, 26–28 May 2010; pp. 33–40.
34. Wang, S. Research on Theories and Methods of Spatial-Temporal Narrative Visualization. Ph.D. Thesis, PLA Information Engineering University, Zhengzhou, China, 2017.

35. Havre, S.; Hetzler, E.; Whitney, P.; Nowell, L. Themeriver: Visualizing thematic changes in large document collections. *IEEE Trans. Vis. Comput. Graph.* **2002**, *8*, 9–20. [CrossRef]
36. Wu, T.; Wu, Y.; Shi, C.; Qu, H.; Cui, W. Piecestack: Toward better understanding of stacked graphs. *IEEE Trans. Vis. Comput. Graph.* **2016**, *22*, 1640–1651. [CrossRef]
37. Dang, T.N.; Wilkinson, L.; Anand, A. Stacking graphic elements to avoid over-plotting. *IEEE Trans. Vis. Comput. Graph.* **2010**, *16*, 1044–1052. [CrossRef]
38. Tominski, C.; Andrienko, N.; Andrienko, N.; Andrienko, N. Stacking-based visualization of trajectory attribute data. *IEEE Trans. Vis. Comput. Graph.* **2012**, *18*, 2565–2574. [CrossRef]
39. Andrienko, G.; Andrienko, N.; Schumann, H.; Tominski, C. Visualization of trajectory attributes in space-time cube and trajectory wall. In *Cartography from Pole to Pole: Selected Contributions to the xxvith International Conference of the Ica, Dresden 2013*; Buchroithner, M., Prechtel, N., Burghardt, D., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 157–163.
40. Andrienko, N.; Andrienko, G. Visual analytics of movement: An overview of methods, tools and procedures. *Inf. Vis.* **2013**, *12*, 3–24. [CrossRef]
41. Du, F.; Zhu, A.X.; Qi, F. Interactive visual cluster detection in large geospatial datasets based on dynamic density volume visualization. *Geocarto Int.* **2016**, *31*, 597–611. [CrossRef]
42. Scheepens, R.; Willems, N.; Wetering, H.V.D.; Andrienko, G.; Andrienko, N.; Wijk, J.J.V. Composite density maps for multivariate trajectories. *IEEE Trans. Vis. Comput. Graph.* **2011**, *17*, 2518–2527. [CrossRef] [PubMed]
43. Scheepens, R.; Hurter, C.; Wetering, H.V.D.; Wijk, J.J.V. Visualization, selection, and analysis of traffic flows. *IEEE Trans. Vis. Comput. Graph.* **2016**, *22*, 379–388. [CrossRef] [PubMed]
44. Demšar, U.; Virrantaus, K. Space–time density of trajectories: Exploring spatio-temporal patterns in movement data. *Int. J. Geogr. Inf. Sci.* **2010**, *24*, 1527–1542. [CrossRef]
45. Li, C.; Baciu, G.; Han, Y. Streammap: Smooth dynamic visualization of high-density streaming points. *IEEE Trans. Vis. Comput. Graph.* **2018**, *24*, 1381–1393. [CrossRef]
46. Rothlisberger, D.; Nierstrasz, O.; Ducasse, S.; Pollet, D.; Robbes, R. Supporting task-oriented navigation in ides with configurable heatmaps. In Proceedings of the IEEE International Conference on Program Comprehension, Vancouver, BC, Canada, 17–19 May 2009; pp. 253–257.
47. Liu, S.; Pu, J.; Luo, Q.; Qu, H.; Ni, L.M.; Krishnan, R. Vait: A visual analytics system for metropolitan transportation. *IEEE Trans. Intell. Transp. Syst.* **2013**, *14*, 1586–1596. [CrossRef]
48. Chen, Y.; Tu, L. Density-based clustering for real-time stream data. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, CA, USA, 12–15 August 2007; pp. 133–142.
49. Babcock, B.; Datar, M.; Motwani, R.; O'Callaghan, L. Maintaining variance and k-medians over data stream windows. In Proceedings of the Twenty-Second ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, San Diego, CA, USA, 9–12 June 2003; pp. 234–243.
50. Chae, J.; Thom, D.; Jang, Y.; Kim, S.; Ertl, T.; Ebert, D.S. Public behavior response analysis in disaster events utilizing visual analytics of microblog data. *Comput. Graph.* **2014**, *38*, 51–60. [CrossRef]
51. Wang, S.; Xu, Z.; Zhang, J.; Du, M. A reverse rendering method of heatmap. *J. Geo-Inf. Sci.* **2018**, *20*, 515–522.
52. Li, C.; Baciu, G.; Han, Y. Interactive visualization of high density streaming points with heat-map. In Proceedings of the 2014 International Conference on Smart Computing, Hong Kong, China, 3–5 November 2014; pp. 145–149.
53. Huang, X.; Zhao, Y.; Ma, C.; Yang, J.; Ye, X.; Zhang, C. Trajgraph: A graph-based visual analytics approach to studying urban network centralities using taxi trajectory data. *IEEE Trans. Vis. Comput. Graph.* **2016**, *22*, 160–169. [CrossRef]
54. Andrienko, N.; Andrienko, G. Spatial generalization and aggregation of massive movement data. *IEEE Trans. Vis. Comput. Graph.* **2011**, *17*, 205–219. [CrossRef] [PubMed]
55. Andrienko, N.; Andrienko, G.; Rinzivillo, S. Exploiting spatial abstraction in predictive analytics of vehicle traffic. *Isprs Int. J. Geo-Inf.* **2015**, *4*, 591–606. [CrossRef]
56. Andrienko, N.; Andrienko, G. A visual analytics framework for spatio-temporal analysis and modelling. *Data Min. Knowl. Discov.* **2013**, *27*, 55–83. [CrossRef]
57. Saito, T.; Miyamura, H.N.; Yamamoto, M.; Saito, H.; Hoshiya, Y.; Kaseda, T. Two-tone pseudo coloring: Compact visualization for one-dimensional data. In Proceedings of the IEEE Symposium on Information Visualization, 2005, INFOVIS 2005, Minneapolis, MN, USA, 23–25 October 2005; pp. 173–180.

58. Wood, Z.; Galton, A. A taxonomy of collective phenomena. *Appl. Ontol.* **2009**, *4*, 267–292.
59. Wood, Z.; Galton, A. Classifying collective motion. In *The Behaviour Monitoring and Interpretation—BMI*; Gottfried, B., Aghajan, H., Eds.; IOS press: Amsterdam, The Netherlands, 2009; Volume 3, pp. 129–155.
60. Galton, A. Zooming in on collective motion. *Pharmacol. Res.* **2001**, *43*, 241–244.
61. Wood, Z.M. Detecting and Identifying Collective Phenomena within Movement Data. Ph.D. Thesis, University of Exeter, Exeter, UK, 2011.
62. Giardina, I. Collective behavior in animal groups: theoretical models and empirical studies. *HFSP J.* **2008**, *2*, 205–219. [CrossRef] [PubMed]
63. Nagy, M.; Ákos, Z.; Biro, D.; Vicsek, T. Hierarchical group dynamics in pigeon flocks. *Nature* **2010**, *464*, 890–893. [CrossRef] [PubMed]
64. Laube, P.; Imfeld, S.; Weibel, R. Discovering relative motion patterns in groups of moving point objects. *Int. J. Geogr. Inf. Sci.* **2005**, *19*, 639–668. [CrossRef]
65. Andrienko, N.; Andrienko, G.; Barrett, L.; Dostie, M.; Henzi, P. Space transformation for understanding group movement. *IEEE Trans. Vis. Comput. Graph.* **2013**, *19*, 2169–2178. [CrossRef]
66. Laney, D. *3-D Data Management: Controlling Data Volume, Velocity, and Variety*; META Group: Terni, Italy, 2001.
67. Sagiroglu, S.; Sinanc, D. Big data: A review. In Proceedings of the 2013 International Conference on Collaboration Technologies and Systems (CTS), San Diego, CA, USA, 20–24 May 2013; pp. 42–47.
68. Figueroa, A. Exploring effective features for recognizing the user intent behind web queries. *Comput. Ind.* **2015**, *68*, 162–169. [CrossRef]
69. Alpaydin, E. *Introduction to Machine Learning*; The MIT Press: Cambridge, MA, USA, 2010; p. 584.
70. Cunningham, J.P.; Ghahramani, Z. Linear dimensionality reduction: Survey, insights, and generalizations. *J. Mach. Learn. Res.* **2015**, *16*, 2859–2900.
71. Pawliczek, P.; Dzwinel, W. Interactive data mining by using multidimensional scaling. *Procedia Comput. Sci.* **2013**, *18*, 40–49. [CrossRef]
72. Moon, K.R.; van Dijk, D.; Wang, Z.; Gigante, S.; Burkhardt, D.; Chen, W.; van den Elzen, A.; Hirn, M.J.; Coifman, R.R.; Ivanova, N.B.; et al. Visualizing transitions and structure for biological data exploration. *bioRxiv* **2018**.
73. Elisa, P.D.S.A. Multidimensional Projection Visualization: Control-points Selection and Inverse Projection Exploration. Ph.D. Thesis, University of Calgary, Calgary, AB, Canada, 2016.
74. Lehmann, D.J.; Theisel, H. Orthographic star coordinates. *IEEE Trans. Vis. Comput. Graph.* **2013**, *19*, 2615–2624. [CrossRef] [PubMed]
75. Kuhn, A.; Lindow, N.; Günther, T.; Wiebel, A.; Theisel, H.; Hege, H.-C. Trajectory density projection for vector field visualization. In Proceedings of the EuroVis 2013, Leipzig, Germany, 17 June 2013; pp. 31–35.
76. Van den Elzen, S.; Holten, D.; Blaas, J.; van Wijk, J.J. Reducing snapshots to points: A visual analytics approach to dynamic network exploration. *IEEE Trans. Vis. Comput. Graph.* **2016**, *22*, 1–10. [CrossRef] [PubMed]
77. Liu, X.; Gong, L.; Gong, Y.; Liu, Y. Revealing travel patterns and city structure with taxi trip data. *J. Transp. Geogr.* **2015**, *43*, 78–90. [CrossRef]
78. Lou, X.; Liu, S.; Wang, T. Fanlens: A visual toolkit for dynamically exploring the distribution of hierarchical attributes. In Proceedings of the 2008 IEEE Pacific Visualization Symposium, Kyoto, Japan, 5–7 March 2008; pp. 151–158.
79. Shneiderman, B. Tree visualization with tree-maps: A 2-d space-filling approach. *ACM Trans. Graph.* **1991**, *11*, 92–99. [CrossRef]
80. Wood, J.; Dykes, J. Spatially ordered treemaps. *IEEE Trans. Vis. Comput. Graph.* **2008**, *14*, 1348–1355. [CrossRef]
81. Stasko, J.; Zhang, E. Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In Proceedings of the IEEE Symposium on Information Visualization 2000. INFOVIS 2000. Proceedings, Salt Lake City, UT, USA, 9–10 October 2000; pp. 57–65.
82. Wu, W.; Xu, J.; Zeng, H.; Zheng, Y.; Qu, H.; Ni, B.; Yuan, M.; Ni, L.M. Telcovis: Visual exploration of co-occurrence in urban human mobility based on telco data. *IEEE Trans. Vis. Comput. Graph.* **2016**, *22*, 935–944. [CrossRef]

83. Bernard, J.; Wilhelm, N.; Krüger, B.; May, T.; Schreck, T.; Kohlhammer, J. Motionexplorer: Exploratory search in human motion capture data based on hierarchical aggregation. *IEEE Trans. Vis. Comput. Graph.* **2013**, *19*, 2257–2266. [CrossRef]
84. Zheng, C. A Visual Analysis System with Large Scale Taxi Origin Destination Data. Master's Thesis, Zhejiang University of Technology, Hangzhou, China, 2015.
85. Wang, Z.; Lu, M.; Yuan, X.; Zhang, J.; Wetering, H.V.D. Visual traffic jam analysis based on trajectory data. *IEEE Trans. Vis. Comput. Graph.* **2013**, *19*, 2159–2168. [CrossRef] [PubMed]
86. Inselberg, A. The plane with parallel coordinates. *Vis. Comput.* **1985**, *1*, 69–91. [CrossRef]
87. Wegman, E.J. Hyperdimensional data analysis using parallel coordinates. *J. Am. Stat. Assoc.* **1990**, *85*, 664–675. [CrossRef]
88. Inselberg, A. *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*; Springer-Verlag: New York, NY, USA, 2009; p. 554.
89. Itoh, T.; Kumar, A.; Klein, K.; Kim, J. High-dimensional data visualization by interactive construction of low-dimensional parallel coordinate plots. *J. Vis. Lang. Comput.* **2017**, *43*, 1–13. [CrossRef]
90. Guo, D.; Zhu, X.; Jin, H.; Gao, P.; Andris, C. Discovering spatial patterns in origin-destination mobility data. *Trans. GIS* **2012**, *16*, 411–429. [CrossRef]
91. Holten, D.; van Wijk, J.J. Evaluation of cluster identification performance for different pcp variants. In Proceedings of the 12th Eurographics/IEEE - VGTC Conference on Visualization, Bordeaux, France, 9–11 June 2010; pp. 793–802.
92. Yuan, X.; Guo, P.; Xiao, H.; Zhou, H.; Qu, H. Scattering points in parallel coordinates. *IEEE Trans. Vis. Comput. Graph.* **2009**, *15*, 1001–1008. [CrossRef]
93. Claessen, J.H.; van Wijk, J.J. Flexible linked axes for multivariate data visualization. *IEEE Trans. Vis. Comput. Graph.* **2011**, *17*, 2310–2316. [CrossRef]
94. Elmqvist, N.; Fekete, J.D. Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE Trans. Vis. Comput. Graph.* **2010**, *16*, 439–454. [CrossRef]
95. Zhou, H.; Yuan, X.; Qu, H.; Cui, W.; Chen, B. Visual clustering in parallel coordinates. *Comput. Graph. Forum* **2010**, *27*, 1047–1054. [CrossRef]
96. Zhou, L.; Weiskopf, D. Indexed-points parallel coordinates visualization of multivariate correlations. *IEEE Trans. Vis. Comput. Graph.* **2018**, *24*, 1997–2010. [CrossRef]
97. Kandogan, E. Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions. In Proceedings of the IEEE Information Visualization Symposium, Late Breaking Hot Topics, Durham, NC, USA, 19 October 1998; pp. 9–12.
98. Cooprider, N.D.; Burton, R.P. Extension of star coordinates into three dimensions. In Proceedings of the Electronic Imaging 2007, San Jose, CA, USA, 29 January 2007; p. 10.
99. Tan, S.C.; Tan, J. Lost in translation: The fundamental flaws in star coordinate visualizations. *Procedia Comput. Sci.* **2017**, *108*, 2308–2312. [CrossRef]
100. Tan, S.C.; Tan, J. Blind spots in star coordinate visualization: Analysis and correction. *Pattern Recognit. Lett.* **2018**, *106*, 7–12. [CrossRef]
101. Chen, K.; Liu, L. Vista: Validating and refining clusters via visualization. *Inf. Vis.* **2004**, *3*, 257–270. [CrossRef]
102. Murray, P.; Forbes, A. Stretchplot: Interactive visualization of multi-dimensional trajectory data. In Proceedings of the 2014 IEEE Conference on Visual Analytics Science and Technology (VAST), Paris, France, 25–31 October 2014; pp. 261–262.
103. Chen, K. Optimizing star-coordinate visualization models for effective interactive cluster exploration on big data. *Intell. Data Anal.* **2014**, *18*, 117–136. [CrossRef]
104. Zhao, Y.; Peng, Y.; Huang, W.; Li, Y.; Zhou, F.; Liao, Z.; Zhang, K. A collaborative visual analytics of trajectory and transaction data for digital forensics: Vast 2014 mini-challenge 2: Award for outstanding visualization and analysis. In Proceedings of the 2014 IEEE Conference on Visual Analytics Science and Technology (VAST), Paris, France, 25–31 October 2014; pp. 371–372.
105. Ferreira, N.; Klosowski, J.T.; Scheidegger, C.E.; Silva, C.T. Vector field k-means: Clustering trajectories by fitting multiple vector fields. In Proceedings of the 15th Eurographics Conference on Visualization, Leipzig, Germany, 17–21 June 2013; pp. 201–210.

106. Enriquez, M.; Kurcz, C. A simple and robust flow detection algorithm based on spectral clustering. In Proceedings of the International Conference on Research in Air Transportation, Berkeley, CA, USA, 22–25 May 2012.
107. Salaun, E.; Gariel, M.; Vela, A.E.; Feron, E. Aircraft proximity maps based on data-driven flow modeling. *J. Guid. Control. Dyn.* **2012**, *35*, 563–577. [CrossRef]
108. Andrienko, G.; Andrienko, N.; Fuchs, G.; Garcia, J.M.C. Clustering trajectories by relevant parts for air traffic analysis. *IEEE Trans. Vis. Comput. Graph.* **2018**, *24*, 34–44. [CrossRef]
109. Rinzivillo, S.; Pedreschi, D.; Nanni, M.; Giannotti, F.; Andrienko, N.; Andrienko, G. Visually driven analysis of movement data by progressive clustering. *Inf. Vis.* **2008**, *7*, 225–239. [CrossRef]
110. Ramos, A.M.; Sprenger, M.; Wernli, H.; Durán-Quesada, A.M.; Lorenzo, M.N.; Gimeno, L. A new circulation type classification based upon lagrangian air trajectories. *Front. Earth Sci.* **2014**, *2*. [CrossRef]
111. Wu, X. Marker Clusterer. Available online: https://github.com/googlemaps/js-marker-clusterer (accessed on 25 October 2018).
112. Hurter, C.; Ersoy, O.; Telea, A. Graph bundling by kernel density estimation. *Comput. Graph. Forum* **2012**, *31*, 865–874. [CrossRef]
113. Holten, D.; Van Wijk, J.J. Force-directed edge bundling for graph visualization. *Comput. Graph. Forum* **2009**, *28*, 983–990. [CrossRef]
114. Ersoy, O.; Hurter, C.; Paulovich, F.; Cantareiro, G.; Telea, A. Skeleton-based edge bundling for graph visualization. *IEEE Trans. Vis. Comput. Graph.* **2011**, *17*, 2364–2373. [CrossRef] [PubMed]
115. Lambert, A.; Bourqui, R.; Auber, D. Winding roads: Routing edges into bundles. *Comput. Graph. Forum* **2010**, *29*, 853–862. [CrossRef]
116. Cui, W.; Zhou, H.; Qu, H.; Wong, P.C.; Li, X. Geometry-based edge clustering for graph visualization. *IEEE Trans. Vis. Comput. Graph.* **2008**, *14*, 1277–1284. [CrossRef] [PubMed]
117. Hurter, C.; Ersoy, O.; Telea, A. Smooth bundling of large streaming and sequence graphs. In Proceedings of the 2013 IEEE Pacific Visualization Symposium (PacificVis), Sydney, Australia, 27 February–1 March 2013; pp. 41–48.
118. Klein, T.; van der Zwan, M.; Telea, A. Dynamic multiscale visualization of flight data. In Proceedings of the 2014 International Conference on Computer Vision Theory and Applications (VISAPP), Lisbon, Portugal, 5–8 January 2014; pp. 104–114.
119. Hurter, C.; Conversy, S.; Gianazza, D.; Telea, A.C. Interactive image-based information visualization for aircraft trajectory analysis. *Transp. Res. Part C* **2014**, *47*, 207–227. [CrossRef]
120. Landesberger, T.V.; Brodkorb, F.; Roskosch, P.; Andrienko, N.; Andrienko, G.; Kerren, A. Mobilitygraphs: Visual analysis of mass mobility dynamics via spatio-temporal graphs and clustering. *IEEE Trans. Vis. Comput. Graph.* **2016**, *22*, 11–20. [CrossRef] [PubMed]
121. Jarrell, S.B. *Basic Statistics (Special pre-publication ed.)*; Wm. C. Brown Pub: Dubuque, Iowa, 1994.
122. Elmqvist, N.; Dragicevic, P.; Fekete, J.D. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Trans. Vis. Comput. Graph.* **2008**, *14*, 1148–1539. [CrossRef] [PubMed]
123. Ahlberg, C.; Shneiderman, B. Visual information seeking: Tight coupling of dynamic query filters with starfield displays. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Boston, MA, USA, 24–28 April 1994; pp. 313–317.
124. Microsoft Office Online: Present Your Data in a Bubble Chart. Available online: https://support.office.com/en-us/article/present-your-data-in-a-bubble-chart-424d7bda-93e8-4983-9b51-c766f3e330d9 (accessed on 16 August 2015).
125. Zhang, H.-X.; Sheng, F.-F.; Xu, P.-Y.; Tang, Y. Visualizing user characteristics based on mobile device log data. *Ruan Jian Xue Bao/J. Softw.* **2016**, *27*, 1174–1187.
126. Wang, W.B.; Huang, M.L.; Nguyen, Q.V.; Huang, W.; Zhang, K.; Huang, T.-H. Enabling decision trend analysis with interactive scatter plot matrices visualization. *J. Vis. Lang. Comput.* **2016**, *33*, 13–23. [CrossRef]
127. Chen, H.; Engle, S.; Joshi, A.; Ragan, E.D.; Yuksel, B.F.; Harrison, L. Using animation to alleviate overdraw in multiclass scatterplot matrices. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, Montreal, QC, Canada, 21–26 April 2018; pp. 1–12.
128. Tobler, W.R. A model of geographical movement. *Geogr. Anal.* **1981**, *13*, 1–20. [CrossRef]
129. Doantam, P.; Ling, X.; Yeh, R.; Hanrahan, P. Flow map layout. In Proceedings of the IEEE Symposium on Information Visualization (INFOVIS 2005), Minneapolis, MN, USA, 23–25 October 2005; pp. 219–224.

130. Selassie, D.; Heller, B.; Heer, J. Divided edge bundling for directional network data. *IEEE Trans. Vis. Comput. Graph.* **2011**, *17*, 2354–2363. [CrossRef] [PubMed]
131. Guo, D.; Zhu, X. Origin-destination flow data smoothing and mapping. *IEEE Trans. Vis. Comput. Graph.* **2014**, *20*, 2043–2052. [CrossRef] [PubMed]
132. Andrienko, G.; Andrienko, N.; Fuchs, G.; Wood, J. Revealing patterns and trends of mass mobility through spatial and temporal abstraction of origin-destination movement data. *IEEE Trans. Vis. Comput. Graph.* **2017**, *23*, 2120–2136. [CrossRef] [PubMed]
133. Guo, D. Visual analytics of spatial interaction patterns for pandemic decision support. *Int. J. Geogr. Inf. Sci.* **2007**, *21*, 859–877. [CrossRef]
134. Wood, J.; Dykes, J.; Slingsby, A. Visualisation of origins, destinations and flows with od maps. *Cartogr. J.* **2010**, *47*, 117–129. [CrossRef]
135. Yan, J.; Thill, J.-C. Visual data mining in spatial interaction analysis with self-organizing maps. *Environ. Plan. B Plan. Des.* **2009**, *36*, 466–486. [CrossRef]
136. Voorhees, A.M. A general theory of traffic movement. *Transportation* **2013**, *40*, 1105–1116. [CrossRef]
137. Zeng, W.; Fu, C.W.; Müller Arisona, S.; Erath, A.; Qu, H. Visualizing waypoints-constrained origin-destination patterns for massive transportation data. *Comput. Graph. Forum* **2015**, *35*, 95–107. [CrossRef]
138. Lu, M.; Liang, J.; Wang, Z.; Yuan, X. Exploring od patterns of interested region based on taxi trajectories. *J. Vis.* **2016**, *19*, 811–821. [CrossRef]
139. Andrienko, G.; Andrienko, N. Spatio-temporal aggregation for visual analysis of movements. In Proceedings of the 2008 IEEE Symposium on Visual Analytics Science and Technology, Columbus, OH, USA, 19–24 October 2008; pp. 51–58.
140. Wilkinson, L.; Friendly, M. The history of the cluster heat map. *Am. Stat.* **2009**, *63*, 179–184. [CrossRef]
141. Slingsby, A.; Kelly, M.; Dykes, J.; Wood, J. Od maps for studying historical internal migration in ireland. In Proceedings of the IEEE Conference on Information Visualization (InfoVis), Seattle, WA, USA, 14–19 October 2012.
142. Kim, S.; Jeong, S.; Woo, I.; Jang, Y.; Maciejewski, R.; Ebert, D.S. Data flow analysis and visualization for spatiotemporal statistical data without trajectory information. *IEEE Trans. Vis. Comput. Graph.* **2018**, *24*, 1287–1300. [CrossRef]
143. Fuchs, R.; Hauser, H. Visualization of multi-variate scientific data. *Comput. Graph. Forum* **2009**, *28*, 1670–1690. [CrossRef]
144. Post, F.H.; Vrolijk, B.; Hauser, H.; Laramee, R.S.; Doleisch, H. The state of the art in flow visualisation: Feature extraction and tracking. *Comput. Graph. Forum* **2010**, *22*, 775–792. [CrossRef]
145. Zou, Y.; Chen, Y.; He, J.; Pang, G.; Zhang, K. 4d time density of trajectories: Discovering spatiotemporal patterns in movement data. *Int. J. Geo-Inf.* **2018**, *7*, 212. [CrossRef]
146. Kjellin, A.; Pettersson, L.W.; Seipel, S.; Lind, M. Evaluating 2d and 3d visualizations of spatiotemporal information. *ACM Trans. Appl. Percept.* **2008**, *7*, 1–23. [CrossRef]
147. Yang, J.; Ward, M.O.; Rundensteiner, E.A. Interring: An interactive tool for visually navigating and manipulating hierarchical structures. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis'02)*; IEEE Computer Society: Washington, DC, USA, 2002; p. 77.
148. Theisel, H. Higher order parallel coordinates. In Proceedings of the 2000 Conference on Vision Modeling and Visualization, Saarbrücken, Germany, 22–24 November 2000; pp. 415–420.
149. Li, J.; Martens, J.-B.; van Wijk, J.J. Judging correlation from scatterplots and parallel coordinate plots. *Inf. Vis.* **2010**, *9*, 13–30. [CrossRef]
150. Hoffman, P.; Grinstein, G.; Marx, K.; Grosse, I.; Stanley, E. DNA visual and analytic data mining. In Proceedings of the 8th conference on Visualization '97, Phoenix, AZ, USA, 24 October 1997; pp. 437–441.
151. Hoffman, P.E. Table Visualizations: A Formal Model and its Applications. Ph.D. Thesis, University of Massachusetts Lowell, Lowell, MA, USA, 2000.
152. Kandogan, E. Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 26–29 August 2001; pp. 107–116.
153. Chen, K.; Liu, L. Ivibrate: Interactive visualization-based framework for clustering large datasets. *ACM Trans. Inf. Syst.* **2006**, *24*, 245–294. [CrossRef]

154. Zheng, Y. Methodologies for cross-domain data fusion: An overview. *IEEE Trans. Big Data* **2015**, *1*, 16–34. [CrossRef]

155. Labrinidis, A.; Jagadish, H.V. Challenges and opportunities with big data. *Proc. VLDB Endow* **2012**, *5*, 2032–2033. [CrossRef]