

Research on Improved Visualization Method of Space Time Cube

Jing Sun, Qingyun Huang, and Huiqun Zhao

Abstract—Spatial temporal data refers to data with geographical location and time label. It has the characteristics of multi-source, massive quantity and fast update. It is a typical big data type. Spatial temporal data analysis is one of the core issues in the field of big data research. In order to better demonstrate the process and results of spatial temporal data analysis, visual processing has become one of the important ways of analysis. The analysis of spatial temporal big data through visualization technology can provide insight into the overall picture and main features of big data. However, when using the visualization technology to analyze large-scale spatial temporal data, the characteristics of spatial temporal big data are not considered, so there are often line-intensive and overlapping coverage problems in the visualization results. This paper proposes an improved space time cube visualization method to solve the above problems. First, cluster the spatial temporal data and then use the space time cube visualization method to display the clustered data. The clustering algorithm used is the sub-trajectory clustering. The experimental results show that the improved space time cube visualization technology has obvious visual effects and clear global features.

Index Terms—Spatial temporal data, visualization, space time cube, sub-trajectory clustering.

I. INTRODUCTION

With the development of computer network technology and data acquisition technology, the application of the Internet, geographic information system, social network and other fields has deepened, resulting in a phenomenon of mixed types of data. These mixed data are continuously aggregated, resulting in a kind of the new data type ----- big data. Such as: text data, video data, map data, spatial temporal data. Among them, spatial temporal data is a typical representative of such data. It contains data of geographical location, time label and other attributes [1]. More than 80% of spatial temporal data is related to geographical location in the real world [2], and the spatial temporal data is rich in content. It can be used in many application areas such as traffic management and user behavior analysis, thus stimulating the growing demand for spatial temporal data applications. However, the constant aggregation of spatial temporal data poses many challenges for data processing. For example, the point of spatial point

and line-to-space data visualization is not clear enough.

Data visualization is a science and technology about the visual representation of data. The visual representation of this data is defined as a kind of information extracted in a summary form, including various attributes and variables of the corresponding information unit. The visualization of spatial temporal data can often explore the potential links and developmental changes in data. The spatial temporal data visualization method is used to visualize the time and space dimensions and the related information object attributes, and to display the patterns and laws closely related to time and space, which can analyze the spatial temporal data well.

The main purpose of spatial temporal data visualization is to be able to process data relationships in different periods and visualize the time dimension in a visual form, so that the trend of geographic goals is clear at a glance. Spatial temporal data visualization can express and interpret spatial temporal and the evolution process of various geographical phenomena, and further predict and simulate its changing development trend by analyzing its development law. Analyze the visualization results and then filter out the useful information in the data, mine the data behind the information, and use visual tools to represent the content covered by the data [3].

At present, typical spatial temporal data visualization methods are flow map and space-time cube. Flow map is a fusion of time information flow and map. Space-time cube visualize time, space and events in three dimensions. Both of them have large-scale spatial temporal data items that cause line intensive and overlapping coverage problems, which is one of the main problems in the visualization of spatial temporal data in the big data environment. Previous studies have used edge bundling, scatter plots and density maps to solve this problem. However, when the dimensions of the null information object attributes are large, the improved visualization method also has some shortcomings.

In order to solve the above problems, this paper optimizes the space-time cube visualization method, divides the trajectories by time period, clusters the target motion trajectory in time and space, and then displays the clustering result in space-time cube technology, which can be effective. Avoid line intensive and overlapping coverage problems, and intuitively analyze attribute information of multiple dimensions of large-scale spatiotemporal data.

This paper is divided into five parts. The first part is the introduction part, which mainly describes the research content and research significance of this paper. The second part introduces the related work of spatial and temporal data visualization. The third section outlines the visualization method and use used in this paper. The fourth section is a visual representation of the experimental data, and the final

Manuscript received January 8, 2019; revised April 3, 2019. This work was supported in part by Natural Science Foundation of China (Grant No: 61672041, 61370051).

The authors are with Computer School, North China University of Technology, Beijing 100144 China (e-mail: sunjing8248@163.com, qingyun_huang@foxmail.com, zhaohq6625@sina.com).

section is the conclusion and future work.

II. RELATED WORKS

This section introduces some research on the visualization of spatial temporal data, including space-time cube and flow map, as well as some improved methods.

Spatial temporal data visualization has two typical methods: flow map and space-time cube. In order to reflect the behavioral changes of information objects as they progress over time and spatial locations, the two typically present data features through the visualization of the properties of the information objects. Charles Joseph Minard used flow map to showcase the export of French red wine in 1864 [4], and the width of the line indicates the number of exports. When the data scale continues to increase, the traditional flow map faces a lot of problems such as crossover and coverage of primitives. In order to solve this problem, Doantam Phan and Kevin Verbeek and others borrowed and merged the edge bundling method in large-scale graph visualization, and bounded the time event stream to optimize the flow map visualization method [5], [6]. In addition, Roeland Scheepens et al. can also solve this problem by optimizing the Flow map by fusing the time event stream based on density calculation [7]. Although the flow map visualization method that integrates other methods can solve the cross-coverage problem, it ignores the visualization of other attributes of the data, and fails to display the three-dimensional information of the data, and has certain limitations.

In order to break through the limitations of the two-dimensional plane, the space time cube visualizes time, space and events in three dimensions. Peuquet DJ uses the space time cube to display and analyze Napoleon's attack on Russia, and can visually display the geographical changes, time changes, personnel changes and special events in the process [8]. But time and space cubes are also facing the intensive mess caused by large-scale data. Rhyne TM et al. combined scatter plots to optimize space-time cubes [9], Tominski *et al* [10]. Merged two-dimensional and three-dimensional visualization methods, introduced a stack graph, and expanded the display space of multi-dimensional attributes in space time cube. The above-mentioned various types of space time cube are suitable for displaying large-scale spatiotemporal data such as urban traffic GPS data and hurricane data. However, when there are many dimensions of the attributes of the empty information object, the three-dimensional also faces the limitation of the display ability.

In order to analyze multidimensional data and discover the relationship between different attributes, scatter plot [11] or parallel coordinates [12] are often used to map the relationship between multidimensional attributes through different colors and shapes. To represent different attributes. However, scatter plots are not suitable for displaying all dimensions at the same time, only for displaying important dimension information. Claessen JHT *et al* [13]. Combine the two visualization techniques of parallel coordinates and scatter plots, and propose a new visualization method, parallel coordinate plots (PCP), which realizes multi-angle

analysis of multidimensional data. Geng *et al* also proposed an improved parallel coordinate method for multidimensional analysis of data [14]. Landesberger *et al* use clustering to simplify line-intensive problems [15]. Kim *et al* employ flow visualization techniques to visualize the spatiotemporal data [16]. Although these methods solve the line intensive problem caused by too many dimensions to a certain extent, they also ignore the attributes of some other dimensions, such as time and space. Aidan Slings by *et al* combined multidimensional parallel axes with traditional map mapping methods to show good results in spatiotemporal data [17].

When analyzing and updating fast spatial temporal big data, you must balance the spatial temporal nature of the data with other properties that the data contains. Therefore, spatial temporal data visualization often needs to be combined with various visualization techniques and clustering algorithms to better represent the multidimensional attributes of data.

III. VISUAL METHODOLOGY

In order to solve the above problems, we propose an improved space time cube visualization method. This section describes the data used in the space time cube visualization method based on the sub-trajectory clustering algorithm and its visualization process.



Fig. 1. Data visualization process.

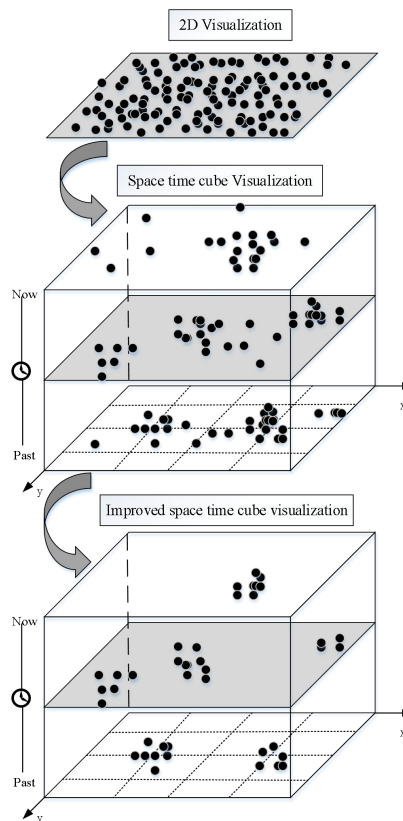


Fig. 2. Space time cube model.

A. Visualization Process

The flow-time data visualization method proposed in this paper is as follows. The spatial temporal data is read from the

database, and then the spatial temporal data is preprocessed. The trajectory is divided into sub-trajectories according to the principle of minimum description length MDL (Minimum Description Length). The clustering method of density clusters these sub-trajectories. Then use the space time cube visualization method to visualize the clustered trajectory, and finally analyze the visualization results. The basic process is shown in Fig. 1.

B. Space Time Cube Model

The space time cube model was first proposed by Hagerstrand. It uses geometric solid graphics to represent the evolution of two-dimensional graphics along the time dimension. It expresses the evolution of the target object over time in the real plane position, and marks the time. At the spatial coordinate point. Given a time position value, the state of the corresponding section can be obtained from the 3D cube, and the process of expressing the 3D space along time can be extended. As shown in Fig. 2, the two-dimensional coordinate axis represents the plane position space of the spatiotemporal trajectory data in the real world, and the one-dimensional time axis represents the change of the position of the target with time.

C. Characteristic Point Extraction

The two most important dimensions and attributes in spatial temporal data are time information and spatial information. Trajectory data is a typical representative of spatial temporal data. The target object will generate trajectory data during the motion process. Therefore, this paper establishes the trajectory data model to visualize and analyze spatial temporal data.

The trajectory data is data information obtained by sampling the motion process of one or more moving objects in a space-time environment, including sampling point position, sampling time, speed, etc., and the sampling point data information constitutes trajectory data according to the sampling sequence. The relevant definitions are as follows:

Definition 1 Trajectory data set. Given a trajectory data set $TR=\{TR_1, TR_2, \dots, TR_{num}\}$, where num represents the total number of trajectories in the set, and any trajectory TR_i in the set is represented as $TR_i=\{P_1, P_2, \dots, P_{ipnum}\}$, where $ipnum$ is the total number of sample points in the i -th trajectory, and any sample point P_k ($1 \leq k \leq ipnum$) in the track has the following form: $P_k \in (P_{traid} \times P_{num} \times P_x \times P_y \times T \times A_1 \times \dots \times A_m)$, which shows at time T , the position of the sample point $pnum$ in the trajectory P_{traid} is (P_x, P_y) , where $1 \leq k \leq num$, and A_i ($1 \leq i \leq m$) is a quantitative or qualitative attribute of the sample point $pnum$, such as speed, corner, etc. .

Definition 2 Characteristic point data set. Dividing a trajectory into multiple segments, the sum of the segmented trajectory is not necessarily the original trajectory, but also the extraction of the original trajectory characteristic, and the ordered characteristic point set representing a trajectory is represented as $CHTR_i=\{P_{chi1}, P_{chi2}, \dots, P_{ichnum}\}$ Where $ichnum$ represents the number of characteristic points of the i -th trajectory, where P_{chi} is the i -th characteristic point of the trajectory.

Definition 3 Sub-trajectory data set. Generating a set of sub-trajectory segments that generate all the trajectories according to the characteristic point sets of all the trajectories,

assuming that there are a total of $lnum$ sub-trajectory segments, the sub-trajectory segment set meanings $SETR=\{L_1, L_2, \dots, L_{lnum}\}$, where $L_i=\langle P_{chi}, P_{chi+1} \rangle$ ($1 \leq k \leq ipnum$), P_{chi}, P_{chi+1} are adjacent characteristic points.

The characteristic point is defined as the point where the behavior change in the trajectory is relatively obvious, and the trajectory structure can be well described, and at the same time, it should have certain simplicity and accuracy. The algorithm for extracting characteristic points is shown in Fig. 3.

```

Algorithm CPE (Characteristic Points Extraction)
INPUT: A set of trajectories  $TR_i = \{P_1, P_2, \dots, P_{ipnum}\}$ 
OUTPUT: A set of characteristic points  $CHTR_i = \{P_{chi1}, \dots, P_{ichnum}\}$ 
01: Add  $P_1$  into the set  $CHTR_i$ ;  $first = P_1$ ;
02:  $start = first$ ;  $current = first.next.next$ ;
03: while( $current.next \neq NULL$ ) do
04:    $cost_{par} = MDL_{par}(P_{start}, P_{current})$ ;
05:    $cost_{nopar} = MDL_{nopar}(P_{start}, P_{current})$ ;
06:   if( $cost_{par} > cost_{nopar}$ ) then
07:     Add  $P_{current}$  into the set  $CHTR_i$ ;
08:   else
09:      $start = start.next$ ;  $current = current.next$ ;
10: Add the  $P_{ipnum}$  into the set  $CHTR_i$ ;
    
```

Fig. 3. The algorithm for extracting characteristic points.

D. Sub-trajectory Segments Clustering Algorithm

This paper uses the spatial distance measurement method commonly used in pattern recognition, which is the weighted sum of the three distances of vertical distance d_{\perp} , parallel distance d_{\parallel} , and angular distance d_{θ} . Suppose the two trajectory segments are $L_i(s_i, e_i)$ and $L_j(s_j, e_j)$, where s_i, s_j, e_i, e_j represent the start and end points of the trajectories L_i and L_j respectively. p_s and p_e represent the projection of s_j and e_j on the trajectory L_i respectively. $L_{\perp 1}, L_{\perp 2}, L_{\parallel 1}, L_{\parallel 2}$ respectively represent the Euclidean distance between the corresponding endpoints in the graph. $\|L_j\|$ represents the length of the trajectory L_j , and θ is the two sub-trajectories Angle ($0^\circ \leq \theta < 180^\circ$). Spatial distance figure between two trajectories is as shown in Fig. 4.

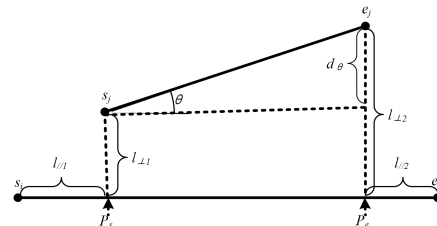


Fig. 4. Spatial distance figure between two trajectories.

Definition 4: The vertical distance is defined as follows:

$$d_{\perp}(L_i, L_j) = \frac{L_{\perp 1}^2 + L_{\perp 2}^2}{L_{\perp 1} + L_{\perp 2}}$$

Definition 5: The parallel distance is defined as follows:

$$d_{\parallel}(L_i, L_j) = \text{MIN}(L_{\parallel 1}, L_{\parallel 2})$$

Definition 6: The angular distance is defined as follows:

$$d_{\theta}(L_i, L_j) = \begin{cases} \|L_j\| \times \sin(\theta), & \text{if } 0^\circ \leq \theta < 90^\circ \\ \|L_j\|, & \text{if } 90^\circ \leq \theta < 180^\circ \end{cases}$$

Definition 7: The weighted sum of the three distances is defined as follows:

$$\text{dist}(L_i, L_j) = \omega_{\perp} d_{\perp}(L_i, L_j) + \omega_{\parallel} d_{\parallel}(L_i, L_j) + \omega_{\theta} d_{\theta}(L_i, L_j)$$

By observing the set of sub-trajectory segments obtained by segmentation of the trajectory, it can be found that the shape of the sub-trajectory segment has irregularities, and the set contains a large amount of noise. Since the DBSCAN algorithm clusters by analyzing the connectivity of regional densities, not only can cluster clusters of arbitrary shapes be found, but also noise interference can be avoided to the utmost in the clustering process. Therefore, the sub-trajectory segments are aggregated by this method. The algorithm needs to set two global parameters, namely the neighborhood radii eps and $minlns$. DBSCAN searches for clusters by filtering the eps neighborhood of each object in the sub-trajectory data set. If the number of objects contained in the eps neighborhood of object p is greater than or equal to $minlns$, a cluster with p as the core object is created. The algorithm then iteratively clusters all objects that are directly reachable from these core objects. This process may involve some density-to-cluster merging. When no new objects are added to any cluster, the process ends. The relevant definitions are given below.

Algorithm Sub-trajectory Segments Clustering

INPUT: A set of trajectory segment $STS = \{L_1, L_2, \dots, L_{lnum}\}$,
Two parameters eps and $minlns$.

OUTPUT: A set of clusters $C = \{C_1, C_2, \dots, C_{cnum}\}$.

```

01: clusterId = 0;
02: Mark all the sub-trajectory segments in STS as unclassified;
03: for each ( $L_i \in STS$ ) do
04:   if ( $L_i$  is unclassified) then
05:     Compute  $N_{\epsilon}(L_i)$ ;
06:     if ( $|N_{\epsilon}(L_i)| > minlns$ ) then
07:       Assign clusterId to  $\forall X \in N_{\epsilon}(L_i)$ ;
08:       Insert  $N_{\epsilon}(L_i) - \{L_i\}$  into the queue Q;
09:       while ( $Q \neq \emptyset$ ) do
10:         Select  $Q_i \in Q$  and compute  $N_{\epsilon}(Q_i)$ ;
11:         if ( $|N_{\epsilon}(Q_i)| > minlns$ ) then
12:           for each ( $X \in N_{\epsilon}(Q_i)$ ) do
13:             if ( $X$  is unclassified or noise) then
14:               Assign clusterId to  $X$ ;
15:             if ( $X$  is unclassified) then
16:               Insert  $X$  into the queue Q;
17:           Remove  $Q_i$  from the queue Q;
18:           clusterId++;
19:         else
20:           Mark  $L_i$  as noise;
21: Output a set of clusters  $C = \{C_1, C_2, \dots, C_{cnum}\}$ 
    
```

Fig. 5. The algorithm for clustering sub-trajectory segments.

Definition 8: Eps neighborhood. The area with the radius eps as the center of the given object p is called the eps neighborhood of the object.

Definition 9: Core object. Within the eps neighborhood of a given object p , if the number of sample points is greater than or equal to $minlns$, the object is said to be the core object.

Definition 10: Direct density is reachable. If the given object q is the core object, and the object p is within the eps

neighborhood of the core object q , then the object p is said to be directly reachable from the object q .

The sub-trajectory segments clustering algorithm is shown in Fig. 5.

IV. CASE STUDIES

To verify the effectiveness of the proposed visualization method, this paper uses two different types of data sets for testing, one is traffic data and the other is sports data, both of which are typical spatial temporal data.

A. Traffic Data Visualization

The traffic data comes from the Microsoft T-Drive project, which contains the trajectory data of more than 10,000 taxis in Beijing in 2008. The data set contains 15 million coordinate points, and the total distance of the trajectory reaches more than 9 million kilometers. The trajectory data of the taxi is shown in Table I.

TABLE I: TAXI TRAJECTORY DATA

Taxi ID	Date time	Longitude	Latitude
1	2008-02-0215:36:08	116.51172	39.92123
1	2008-02-0215:46:08	116.51135	39.93883
...
10357	2008-02-0213:42:25	116.45186	39.93225
...
10357	2008-02-0817:26:51	116.72877	40.01143

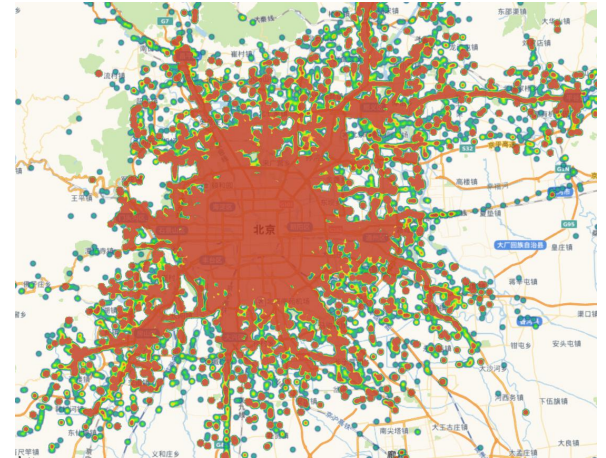


Fig. 6. Taxi trajectory heat map.

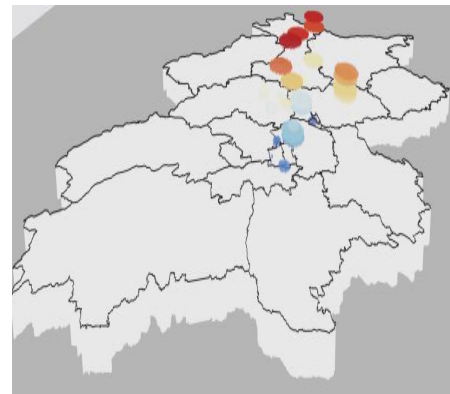


Fig. 7. Taxi trajectory heat map by time.

The heat map shown in Fig. 6 is the taxi trajectory. It can be clearly found that the taxi has a dense running track,

multiple overlapping coverage, and it is impossible to distinguish the congested road sections and areas. In this paper, the sub-trajectory clustering algorithm is used to segment and process the taxi trajectory. As shown in Fig. 7, the area where the taxi is most densely displayed can be clearly seen, and the location and size of the dense area change with time. Fig. 8 shows the most congested areas at a time. The improved visualization method solves the problem of dense track and overlapping coverage, and can quickly locate congestion information according to time.



Fig. 8. Taxi trajectory heat map at a certain moment.

B. Sports Data Visualization

The format of table tennis trajectory data in the quarter-finals of the 2017 Asian Championships in Ding Ning VS Hirano Miyuki is as follows:

Match	Win	Lose	Shot	X	Y
1	0	0	1	561	188
1	0	0	2	-1439	-183
...
5	10	11	11	-703	756
5	10	11	12	1272	-576

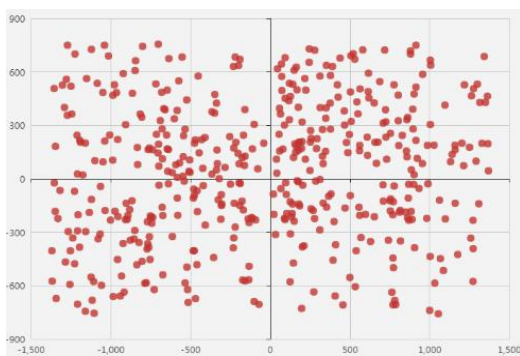


Fig. 9. Table tennis location map.

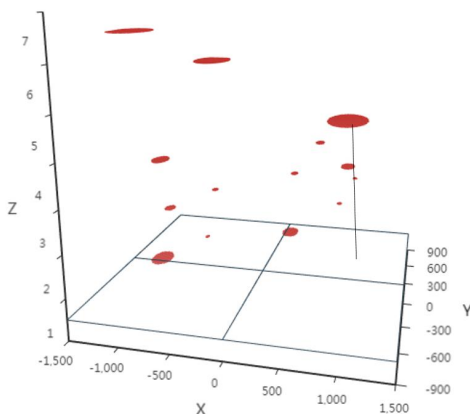


Fig. 10. Table tennis location heat map by time.

As shown in Fig. 9, both players' shots overlap and overlap, and the preferred position of the player's shot is not observed. It is processed by the sub-trajectory clustering algorithm, and the hitting hot spot area of the player is displayed in half. Fig. 10 shows the hot spot area of the top seven hitting position of Hirano Miyuki. The larger the area of the red circle is, the area hits. The more the number of balls, the red box is the preferred hitting position of the Hirano Miyu third board.

V. DISCUSSION

Lessons Learned. The entire visualization process provides us with invaluable experience in analyzing spatial temporal data, and also gives us a deeper understanding of the characteristics of spatiotemporal data. First of all, the new visualization method proposed in this paper can fully display the time and spatial information of spatiotemporal data. Secondly, we also apply this visualization method to the actual scene to analyze the athlete's game data, which can obviously observe the preference hitting position of table tennis players at different times. We provide them with more scientific decisions by Analysis of table tennis match data. Finally, experts from the Sports Science Research Institute of the State Sports General Administration played an important role in this process. They gave many scientific advice and made it easier for us to understand the data.

Limitations. Although the improved visualization method can take into account both time information and spatial information, other dimensions are not fully demonstrated, and the interaction between the system and the user is less.

VI. CONCLUSION AND FUTURE WORK

The spatial temporal data visualization method described in this paper mainly uses sub-trajectory clustering algorithm and space time cube to realize the visualization of spatial temporal data according to time division, so as to preserve the multi-dimensional attribute visualization without causing line intensive.

Future work focuses on the following two aspects:

- (1) Strengthen the correlation analysis of spatial temporal data, and try to analyze the potential law between data
- (2) Combining machine learning with Bayesian networks to predict the evolution of spatial temporal data.

CONFLICT OF INTEREST

We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

AUTHOR CONTRIBUTIONS

Jing Sun analyzed the data; Qingyun Huang and HuiQun Zhao conducted the research and wrote the paper; all authors had approved the final version.

ACKNOWLEDGMENT

The authors would like to thank Dandan Chen from the Sports Science Research Institute of the State Sports General Administration for providing valuable suggestions for this project.

REFERENCES

- [1] L. Ren, Y. Du, S. Ma, X. L. Zhang, and G. Z. Dai, "Visual analytics towards big data," *Journal of Software*, vol. 25, no. 9, pp. 1909-1936, 2014.
- [2] S. Wang, G. Ding, and M. Zhong, "Reflections on spatial data mining under big data," *Journal of Chinese Academy of Electronics*, vol. 8, no. 1, pp. 8-17, 2013.
- [3] Y. Lu, "SOM-based visual analysis model of agricultural spatiotemporal data," Zhejiang Agriculture and Forestry University, 2015.
- [4] W. Tobler, "Experiments in migration mapping by computer," *The American Cartographer*, vol. 14, no. 2, pp. 155-163, 1987.
- [5] D. Phan, L. Xiao, R. Yeh, P. Hanrahan, and T. Winograd, "Flow map layout," in *Proc. the INFOVIS. Los Alamitos*, IEEE Press, 2005, pp. 219-224.
- [6] K. Buchin, B. Speckmann, and K. Verbeek, "Flow map layout via spiral trees," *IEEE Trans. on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2536-2544, 2011.
- [7] R. Scheepens, N. Willems, H. Van de Wetering, G. Andrienko, N. Andrienko, and J. J. van Wijk, "Composite density maps for multivariate trajectories," *IEEE Trans. on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2518-2527, 2011.
- [8] D. J. Peuquet and M. J. Kraak, "Geobrowsing: Creative thinking and knowledge discovery using geographic visualization," *Information Visualization*, vol. 1, no. 1, pp. 80-91, 2002.
- [9] T. M. Rhyne, A. M. MacEachren, and J. Dykes, "Exploring geovisualization," in *Proc. IEEE Computer Graphics and Applications*, 2006, vol. 26, no. 4, pp. 20-21.
- [10] C. Tominski, H. Schumann, G. Andrienko, and N. Andrienko, "Stacking-based visualization of trajectory attribute data," *IEEE Trans. on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2565-2574, 2012.
- [11] C. Ahlberg and B. Shneiderman, "Visual information seeking: Tight coupling of dynamic query filters with star-field displays," in *Proc. the CHI*, New York: ACM Press, 1994, pp. 313-317.
- [12] A. Inselberg and B. Dimsdale, "Parallel coordinates: A tool for visualizing multi-dimensional geometry," in *Proc. the Visualization*, IEEE Press, 1990, pp. 361-378.
- [13] J. H. T. Claessen and J. J. van Wijk, "Flexible linked axes for multi-variate data visualization," *IEEE Trans. on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2310-2316, 2011.
- [14] Z. Geng, Z. Peng, R. S. Laramée, J. C. Roberts, and R. Walker, "Angular histograms: Frequency-Based visualizations for large, high dimensional data," *IEEE Trans. on Visualization and Computer Graphics*, vol. 17, no. 12, p. 25722580, 2011.
- [15] T. V. Landesberger, F. Brodtkorb, P. Roskosch, N. Andrienko, G. Andrienko, and A. Kerren, "Mobility graphs: Visual analysis of mass mobility dynamics via spatio-temporal graphs and clustering," *IEEE Transactions on Visualization & Computer Graphics*, vol. 22, pp. 11-20, 2016.
- [16] S. Kim, S. Jeong, I. Woo, Y. Jang, R. Maciejewski, and D. S. Ebert, "Data flow analysis and visualization for spatiotemporal statistical data without trajectory information," *IEEE Transactions on Visualization and Computer Graphics*, 2017.
- [17] A. Slingsby, J. Dykes, and J. Wood, "Exploring uncertainty in geo-demographics with interactive graphics," *IEEE Trans. on*

Visualization and Computer Graphics, vol. 17, no. 12, pp. 2545-2554, 2011.

Copyright © 2020 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).



Jing Sun was born in January 1968 in Tieling City, Liaoning Province, China. From September 1986 to July 7, she graduated from the Mathematics Department of Liaoning University with a Bachelor of Science degree. From September 1998 to July 2000, she graduated from the Department of Computer Science and Technology of Liaoning University with a master of science degree.

Her research interests are big data analytics, internet of things technology, and software engineering. She has participated in four projects of the National Natural Science Foundation of China and one of the Beijing Academic Innovation Team Projects. She also published more than 50 academic papers.

She is a senior member of the Chinese Computer Society.



Qingyun Huang received the bachelor's degree in computer science and technology from the North China University of Technology, Beijing, China, in 2016. She is currently a graduate student in software engineering at North China University of Technology. Her main research area is data visualization.



Huiqun Zhao was born in September 1960 in Anshan City, Liaoning Province, China. From September 1979 to July 1983, he obtained a bachelor of science degree in mathematics from Jinzhou Normal University. From September 1988 to March 1991, he obtained a master's degree in engineering from the Computer Science Department of Northeastern Institute of Technology. In July 2002, he studied in the Department of Computer Science and Engineering of Northeastern University and obtained a doctorate in engineering.

His research interests are big data analytics, IoT technology and software engineering. He has presided over four projects of the National Natural Science Foundation of China, one project of Beijing Natural Science Foundation, and one project of Beijing Academic Innovation Team. Published more than 100 academic papers.

He is a senior member of the Chinese Computer Society, a member of the software engineering committee of the Chinese computer society, the service computing professional committee, and the professional committee for fault tolerance.