

Hot Spot Analysis of Tourist Attractions Based on Stay Point Spatial Clustering

Yifan Liao*

Abstract

The wide application of various integrated location-based services (LBS social) and tourism application (app) has generated a large amount of trajectory space data. The trajectory data are used to identify popular tourist attractions with high density of tourists, and they are of great significance to smart service and emergency management of scenic spots. A hot spot analysis method is proposed, based on spatial clustering of trajectory stop points. The DBSCAN algorithm is studied with fast clustering speed, noise processing and clustering of arbitrary shapes in space. The shortage of parameters is manually selected, and an improved method is proposed to adaptively determine parameters based on statistical distribution characteristics of data. DBSCAN clustering analysis and contrast experiments are carried out for three different datasets of artificial synthetic two-dimensional dataset, four-dimensional Iris real dataset and scenic track retention point. The experiment results show that the method can automatically generate reasonable clustering division, and it is superior to traditional algorithms such as DBSCAN and k-means. Finally, based on the spatial clustering results of the trajectory stay points, the Getis-Ord G_i^* hotspot analysis and mapping are conducted in ArcGIS software. The hot spots of different tourist attractions are classified according to the analysis results, and the distribution of popular scenic spots is determined with the actual heat of the scenic spots.

Keywords

Analysis, Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Scenic Spot, Spatial Clustering, Stay Point, Trajectory

1. Introduction

The rapid development of technologies, such as mobile Internet and wireless positioning, has led to widespread use of intelligent mobile terminals of various location-based service (LBS) [1], and a large amount of spatial trajectory data has been generated. Spatial data mining technology can obtain hidden knowledge, such as motion characteristics and behavior patterns of moving objects, from the expanding massive trajectory data, and it has been widely used in urban planning, intelligent transportation, environmental monitoring, medical health and public travel [2]. At present, smart tour guide systems and mobile apps such as WeChat can also collect a large amount of visitors' trajectory data or location information in the scenic area. However, it is difficult to recognize and maximize the use of data for many tourist attractions and management departments. Further works are required to explore how to use spatial data mining technology to analyze the data of a large number of tourists' trajectories. Hidden knowledge is

※ This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Manuscript received November 28, 2019; first revision May 22, 2020; accepted June 1, 2020.

Corresponding Author: Yifan Liao (matlab_bysj@126.com)

* School of Information and Mechanical Engineering, Hunan International Economics University, Changsha, China (matlab_bysj@126.com)

obtained, such as popular scenic spots and tourist routes, tourists' similarities and potential interest points. The knowledge could be used to carry out personalized tourism information push and scenic area wisdom management applications. It will gradually become a subject of active research for the tourism industry in the era of "smart tourism" [3-5].

Moving objects database (MOD) can be used to manage trajectory information of vehicles, animals, and other mobile objects. It is crucial to identify efficient methods for tracking the trajectory of an object in real-time, especially for the trajectory data sensed at the mobile object that have to be communicated over a wireless network. A family of tracking protocols have been proposed [6], which enable trading of the communication cost and the storage of trajectory data at a MOD off against the spatial accuracy. With each of these protocols, the MOD manages a simplified trajectory without deviation by more than a certain accuracy bound from the actual movement. Moreover, different protocols enable several trade-offs between computational costs, communication cost, as well as the reduction in the trajectory data. Connection-preserving dead reckoning minimizes the communication cost using dead reckoning, a technique originally designed for tracking an object's current position. Generic remote trajectory simplification (GRTS) further separates tracking of the current position and simplification of the past trajectory, and it can be realized with different line simplification algorithms. Vehicle routing is an important service used by both private individuals and commercial enterprises. Drivers may have different contexts that are characterized by different routing preferences. For example, at different time points of day or weather conditions, drivers may make different routing decisions, such as preferring or avoiding highways. The increasing availability of vehicle trajectory data yields an increasingly rich data foundation for context-aware, preference-based vehicle routing [7]. In the on-line phase, given a context, the corresponding routing preference was identified and used for routing. To achieve efficiency, the preference-based contraction hierarchies are capable of speeding up both off-line learning and on-line routing. Empirical studies with vehicle trajectory data offer novel insights into the properties of the proposed solution.

Clustering is a typical data mining technique that partitions a dataset into multiple subsets of similar objects according to similarity metrics [8,9]. In particular, density-based algorithms can identify clusters of different shapes and sizes while remaining robust to noise objects. DBSCAN (Density-Based Spatial Clustering of Applications with Noise), a representative density-based algorithm, retrieves clusters by defining the density criterion with global parameters, ϵ -distance and MinPts. However, most density-based algorithms, including DBSCAN, cannot obtain clusters correctly, since the density criterion is fixed to the global parameters and misapplied to clusters of varying densities. Although studies have been conducted to determine optimal parameters or to improve clustering performance by using additional parameters and computations, running time for clustering has been significantly prolonged, particularly for large dataset. In recent years, some scholars have carried out researches on mobile object trajectory data and achieved many results. For example, density-based clustering analysis of trajectory data for moving objects on the road was used to identify popular routes in the traffic network, and urban traffic was constructed based on a large amount of taxi historical trajectory data. The density model has been used to predict the traffic congestion in different time periods. In addition, multi-scale analysis was performed by mining the global and process patterns of the starting and ending points of the historical GPS track. Urban moving trajectory pattern mining method is combined with spatial division and road network modeling. Two sequential pattern mining algorithms were used to find the frequent trajectory patterns based on the spatial distribution characteristics of the original moving trajectory on the fuzzy spatial division. In terms of the stop point mining in the motion trajectory, the historical stay points were extracted from the user GPS trajectory information. The density-

based clustering and spatio-temporal data modeling analyses were conducted on the extracted stay points. Furthermore, the user's similarity was explored in geospatial sequence and hierarchy, and the stop point data are combined in the trajectory. The velocity spatiotemporal clustering method was used to determine the interest region of the moving object in the trajectory data [10,11].

Most of these studies are based on mining taxi or volunteer trajectory datasets, extracting the motion feature patterns of moving objects, and finding similarities between users or the relationship between users and geographic regions. However, limited evidence so far focuses on research and application on the historical trajectory data of a large number of tourists in the tourist area, the analysis of popular scenic spots and the recommendation of tourism. Based on the in-depth study of various trajectory data mining algorithms, the improved DBSCAN clustering algorithm is used in this study to cluster the trajectory stay points. Then, the Getis-Ord G_i^* spatial statistical analysis is combined in geographic information system (GIS) to visualize the hotspot distribution as a map. In the hotspot area, the heat level of different attractions is determined based on the quantitative analyses, which is recommended for the popular attractions of the intelligent tour guide app.

2. Stay Point Spatial Clustering Method

The spatial clustering of staying points is a method to extract the staying points from the trajectory data of a large number of tourists. Based on the attributes such as distance, the spatial dataset is divided into different clusters or regions using a specific algorithm, so as to uncover knowledge implicit in the data. At present, there are many spatial clustering algorithms such as k-means, expectation maximization, CURE, DBSCAN, OPTICS, STING and WaveCluster. DBSCAN algorithm, as one of the most popular density-based clustering algorithms, can effectively filter low-density regions in space and identify dense regions of any shape with poly [10,11]. It has advantages of fast speed and ability to deal with noise. Additionally, DBSCAN algorithm is especially suitable for hotspot cluster analysis of such problems as irregular distribution of tourists in tourist attractions.

2.1 Major Definitions

Before performing spatial clustering analysis of trajectory stay points based on DBSCAN algorithm, several main definitions are provided as follows:

Definition 1 (Trajectory): A set of GPS spatial points are connected in chronological order, and it is expressed as follows:

$$T_p = \{p_i | p_i = (x_i, y_i, t_i), i \in N\} \quad (1)$$

where x , y represent the two-dimensional coordinates of the track point, t represents time, and N represents the number of points that make up the track.

Definition 2 (Stay point): A point at which a visitor stays in a geographic area for more than a certain time interval. User retention points can be extracted from the trajectory according to a certain time threshold and distance threshold.

Definition 3 (Hot spot): A region where the staying point is dense, that is, the area where the cluster density exceeds a certain threshold following spatial cluster analysis.

Definition 4 (ε -neighborhood): The ε -neighbor of an object p is the region with p as the core and radius ε , i.e.,

$$N_\varepsilon(p) = \{q \in D \mid \text{dist}(p, q) \leq \varepsilon\} \quad (2)$$

where D is the sample dataset and $\text{dist}(p, q)$ represents the distance between two objects p and q in D .

Definition 5 (Core object): For sample dataset D , object $p \in D$, the neighborhood density threshold $MinPts$ is given, if $|N_\varepsilon(p)| \geq MinPts$, i.e., when the number of sample points in the ε -neighborhood of p is greater than or equal to $MinPts$, the object p is regarded as the core object.

Definition 6 (Direct density reachable): For the sample dataset D , if $q \in N_\varepsilon(p)$, $|N_\varepsilon(p)| \geq MinPts$, i.e., the object q is in the ε -neighborhood of p , and p is the core object, the object q is considered to be directly reachable from the object p .

Definition 7 (Density reachable): For the sample dataset D , if there is a sequence of objects p_1, p_2, \dots, p_n , where $p = p_1$, $q = p_n$, so that p_{i+1} is the direct density reachable from the object p_i ($0 < i < n$), the object q is considered to be density reachable from the object p . The density can be asymmetrical.

Definition 8 (Density connection): For an object o in the sample dataset D , if the object o to the object p and the object q are both reachable, the object p is regarded as density connection to the object q .

Definition 9 (Cluster and noise): Starting from any core object, all objects that are reachable by the object constitute a cluster, and objects that do not belong to any cluster are noisy.

2.2 DBSCAN Algorithm Principle

According to the number of objects contained in a certain area in the cluster space that is no less than a given threshold, the DBSCAN algorithm divides the area with sufficient density into clusters, and the cluster corresponds to the largest set of points connected by density. Unlike the k-means clustering algorithm, the DBSCAN algorithm can automatically determine the number of clusters and discover clusters of any shape in space.

The DBSCAN algorithm flow is described as follows:

- Step 1:** Scan each object p in dataset D in turn. If p has not been accessed, check the ε -neighborhood of p , and obtain the number of objects $|N_\varepsilon(p)|$ in the ε -neighborhood of p . If $|N_\varepsilon(p)|$ is greater than or equal to the specified $MinPts$ value, then p is the core object. p is marked as accessed and a new cluster C is created, p is placed in the new cluster C , and objects in the ε -neighborhood of p are added to the candidate Set N ; if $|N_\varepsilon(p)|$ is less than $MinPts$, p is marked as noise.
- Step 2:** Process each object q in the candidate set N in turn, and process the next object in N if q is marked as accessed. If q is not accessed, check the ε -neighborhood of q , and obtain the number of objects $|N_\varepsilon(q)|$ in the ε -neighborhood of q ; if $|N_\varepsilon(q)| \geq MinPts$, q is the core object and is marked as accessed; if q does not belong to any cluster, then q is added to C . If $|N_\varepsilon(q)| < MinPts$, mark q as noise and remove q from N .

Step 3: Repeat Step 2 to continue checking for objects that are not processed in candidate set N until N is empty.

Step 4: Repeat steps Step 1–Step 3, until all objects have been accessed and are classified into one of the clusters or recognized as noise.

2.3 DBSCAN Algorithm Parameter Adaptive Determination

In the DBSCAN algorithm, two parameters, Eps and $MinPts$, need to be provided in advance, and the clustering result is sensitive to these two parameters. If they are improperly selected, the cluster quality will be degraded. To this end, some scholars have proposed a variety of improved algorithms for adaptive determination of parameters, such as non-parametric kernel density estimation method and differential evolution method, but these improved algorithm mathematical models are complex with relatively low efficiency. Based on the spatial statistical properties of the sample data in this study, the k -nearest neighbor distance matrix and the mathematical expectation method are used to adaptively determine the parameters.

2.3.1 Adaptive determination of Eps

First, the Euclidean distance is calculated from each point in the sample dataset D to other points, a distance distribution matrix $Dist_{n \times n}$ is formed, as shown in Eq. (3):

$$Dist_{n \times n} = \{dist(i, j) | 1 \leq i \leq n, 1 \leq j \leq n\} \quad (3)$$

where n is the number of objects in dataset D , and $dist(i, j)$ is the distance between object i and object j in dataset D .

$Dist_{n \times n}$ is a matrix of n rows and n columns. Each column represents the distance vector of an object to all other objects in D . The distance from the object to itself is 0, so $Dist_{n \times n}$ is a real symmetric matrix with diagonal elements zero. The values of each column element of $Dist_{n \times n}$ are arranged in an ascending order, and the element of the $(k+1)$ -th row can constitute the distance vector $Dist_k$ of each object in the dataset D to the k -th nearest neighbor object. The ascending order of the elements in $Dist_k$ gives the k nearest neighbor distance curve k - $dist$ of dataset D . When k takes different values, different k - $dist$ curves can be plotted, as shown in Fig. 1. The curve corresponding to $k=4$ (thick line identification) is taken as an example. The shapes of the other curves are similar, and they all rise sharply after a steady change, which means that most of the values in $Dist_k$ fall in the dense area corresponding to the gentle section of the curve. The distance is corresponding to the point at which the curvature suddenly changes after the steady rise on the graph, which can be considered as a suitable Eps value. The traditional DBSCAN algorithm assumes that $MinPts=4$, and the value of Eps is determined by observing the threshold value of the noise quantity change descending graph when Eps takes different values. This method requires manual interaction to determine another parameter when one parameter is fixed.

In this study, according to the distribution principle of k - $dist$ curve data, the first derivative of each point is directly obtained in the distance vector $Dist_k$ for the case where the curve distribution is relatively smooth. After removing some extreme values at both ends of the curve, the maximum value of the first derivative is found in the abrupt region after the steady curve rise. The index position is corresponding to the value of the data point number (abscissa), and then the distance value (ordinate) is determined, which is corresponding to the point as a reasonable Eps value where the slope changes the most from

$Dist_k$, as shown in Fig. 2. If the 4-dist curve is not smooth enough, the Gaussian function or the least squares method can also be used to perform curve fitting, and then the fitting function is derived to obtain the curve inflection point.

It is found through experiments that the Eps value calculated by this method is determined by the k -th nearest neighbor distance curve, but when k takes different values such as $[1, 2, \dots, 20]$, the value of Eps does not change significantly as the value of k increases. Therefore, when $k=4$ is selected in this study, the corresponding k -dist curve comes from the adaptive determination parameter Eps .

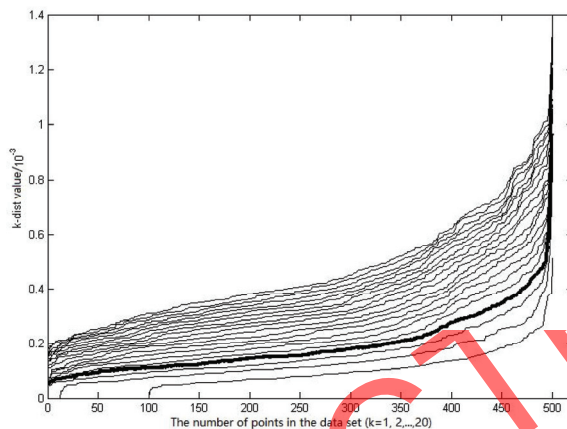


Fig. 1. Example of k -dist distribution curve ($k=1, 2, \dots, 20$).

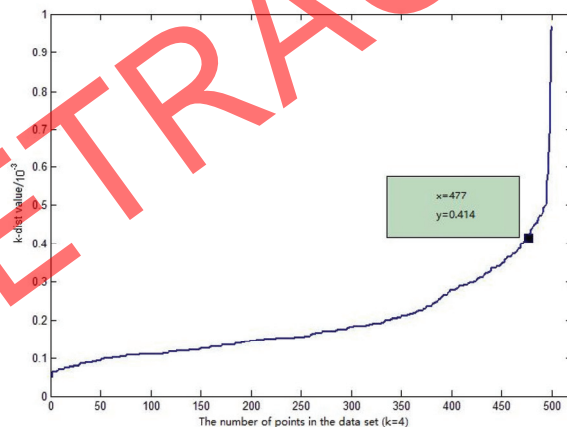


Fig. 2. Determination of Eps ($k=4$) based on the k -dist curve inflection point.

2.3.2 Adaptive determination of $MinPts$

After the Eps is determined, the number of objects in the ϵ -neighborhood of each sample point in the dataset D is calculated in turn. Subsequently, the mathematical expectation is obtained for the number of objects in each sample ϵ -neighborhood in the entire dataset, and the value is taken as the core object of the dataset D . $MinPts$ can be obtained by the optimal value of the number of objects in the ϵ -neighborhood. Its calculation formula is expressed as follows (4):

$$MinPts = \frac{1}{n} \sum_{i=1}^n P_i \tag{4}$$

where P_i is the number of objects in the ε -neighbor of the object, and n is the total number of samples in dataset D .

3. Experimental Results

The DBSCAN algorithm and the parameter adaptive determination method are all conducted in MATLAB. The three types of datasets, such as synthetic data, UCI real test data and track stop point data, are used for DBSCAN cluster analysis and experimental comparisons.

In the early stage, an intelligent tour guide app integrating mobile location service and map navigation function was developed for the Tianya Haijiao Scenic Spot in Sanya, Hainan, China. Upon the running of map navigation function, the user will collect a user location point every 5 seconds or 10 m. In addition to the location on the map in real time, the trajectory space data are composed of these GPS points, which are all stored in the background database. At present, the app has been put into operation, and it has been downloaded and used by many users. The app collects GPS trajectories of tourists visiting the scenic spots in different periods. The track point data include ID, IMEI (unique identification code of mobile phone), acquisition time, longitude, latitude, etc. These attributes are stored in the MySQL database.

Due to the huge number of track points in the database, the spatial trajectory data are acquired during a peak travel period. These data are selected and filtered according to the IMEI code, time and position change information to ensure that only one sample of stop points indicates each visitor within a specified radius (e.g., 20 m). Moreover, they represent the spatial distribution of visitors to different areas of the attraction at certain time points. After sampling of the GPS trajectory, 500 sets of user stay points (StopDS) are extracted. The spatial distribution of the latitude and longitude coordinates of these stay points is normalized by the zscore function in MATLAB.

The k -dist distance distribution graph and the mathematical expectation method are used to statistically analyze the data, and the clustering parameters $Eps=0.1291$, $MinPts=16$ are obtained. The clustering results of the DBSCAN algorithm are shown in Fig. 3(a), and there are 6 clusters. These clusters show different shapes, and the points are dense in each cluster, while those farther away from each cluster are marked as noise. By using the traditional DBSCAN method, $MinPts=4$ is set. If $Eps=0.1291$, the clustering result only presents 4 clusters, as shown in Fig. 3(b).

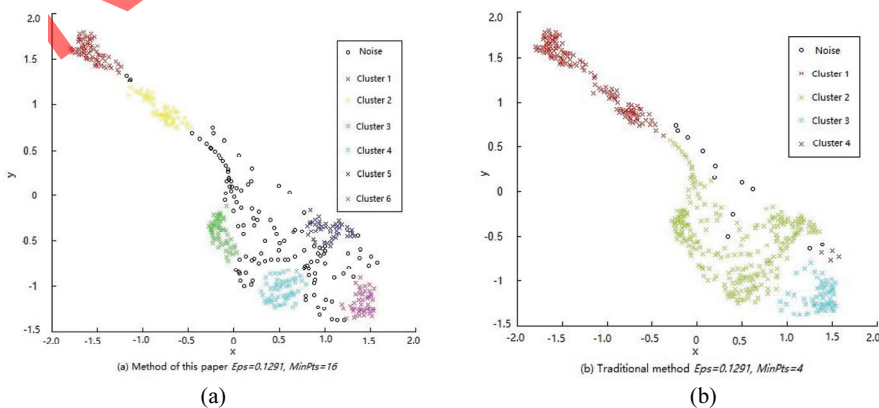


Fig. 3. DBSCAN clustering results of StopDS under different parameters. (a) Clustering results of the method in this article: $Eps=0.1291$, $MinPts=16$. (b) Clustering results of traditional method: $Eps=0.1291$, $MinPts=4$.

Despite the small noise point of the conventional method, the regions are combined with a relatively large density to form fewer clusters, and the dots in some clusters are more dispersed. Because the stop point data of the tourists are not regular in the scenic spot, the shape is not fixed after clustering, but the algorithm can resist the noise and identify the irregularly shaped cluster. The k-means algorithm cannot identify the non-circular shape clusters, and the anti-noise ability is not strong. Therefore, a large amount of noise is classified into clusters, and the clustered point distribution is not tight enough.

In order to reflect the spatial clustering effect more intuitively, the noise is first removed in Fig. 3(a), and then the centroid coordinates and the number of objects of the 6 clusters are calculated, followed by analysis with the ESRI's desktop-side geographic information platform software ArcMap 10.1 software. The clustering results are marked on the scenic electronic map, as shown in Fig. 4. The large dots in the figure represent the center of each cluster, and the indicated number indicates the number of objects in each cluster. The cluster with the highest density displays 89 points, and the smallest cluster presents 46 points.

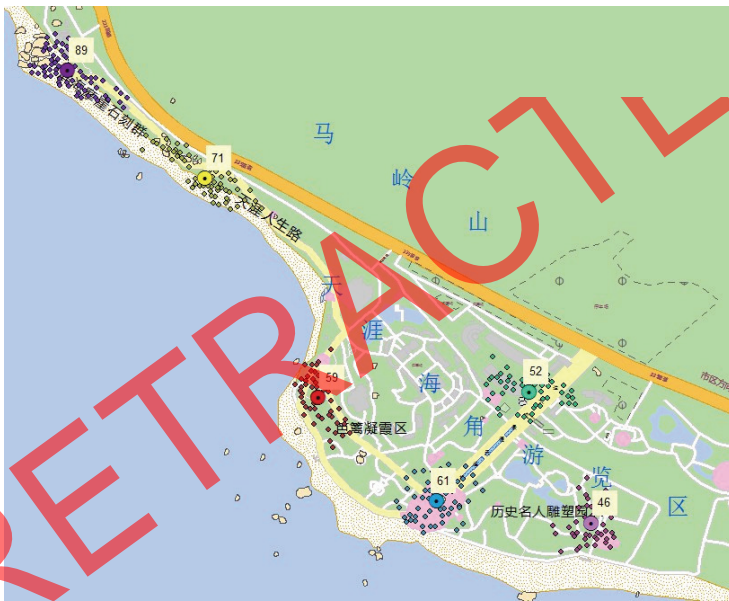


Fig. 4. Spatial distribution of clustering results on scenic maps.

On the basis of the clustering results in Fig. 1(a), a circle is plotted with the center of each cluster as the center and ε as the radius. For the scenic spots falling within the circle, the hot attribute value is set to the cluster containing the number of objects. For attractions out of any clusters, the hot value is set with distance d to the nearest cluster center, according to the following formula:

$$h = \frac{\varepsilon}{d} C \quad (5)$$

where ε represents the neighborhood radius of the cluster, and C is the number of objects of the cluster.

The additional computation is minimized to determine the parameters by using the approximate adaptive ε -distance for each density, while the clusters are found with varying densities, which are what DBSCAN cannot find. Specifically, a new tree structure is given based on a quadtree to define a dataset

density layer. In addition, the approximate adaptive DBSCAN (AA-DBSCAN) and kAA-DBSCAN are proposed, and they have clustering performance similar to those of existing algorithms for finding clusters with varying densities while significantly reducing the running time required to perform clustering [12]. We evaluated the proposed algorithms, AA-DBSCAN and kAA-DBSCAN, via an array of experiments using the state-of-the-art algorithms. Experimental results demonstrate an improvement in clustering performance and reduction in running time of the proposed algorithms.

4. Conclusions

In the present study, DBSCAN algorithm is used to analyze the spatial clustering of trajectory stay points, and an improved method is proposed based on the statistical characteristics of the dataset for determining the neighborhood radius *Eps* and density threshold *MinPts* parameters of DBSCAN algorithm. Spatial aggregation analyses were performed by using three different datasets. The qualitative and quantitative comparative analyses of the clustering results are applied to confirm the effectiveness of the above-mentioned method. This method is simple and efficient, as compared with the method of establishing mathematical model with a nuclear density estimation theory, or other methods of adaptively determining DBSCAN parameters using differential evolution algorithm for mutation, crossover, selection and other optimizing iterations. The time complexity is low. The optimized parameters are used to perform DBSCAN cluster analysis on the tourist trajectory stop point data of a tourist scenic spot. In addition, the Getis-Ord G_i^* hotspot analysis in GIS is used for spatial statistics and mapping on the clustering results. These can visually display the hierarchical distribution of the hotspot area, and the hotspot area is obtained by the analysis. Thus, these are basically consistent with the conclusions given by the scenic area management department.

Global *Eps* and *MinPts* parameters are used for clustering in this study, and it is suitable for datasets with uniform density distribution. In terms of clusters with uneven density in the dataset, the *Eps* values are corresponding to the post-stabilization mutation points, and they may be selected according to the *k* nearest neighbor distance curve distribution. Furthermore, different *Eps* and *MinPts* parameters are used for clustering in different data regions. Thereby, spatial clustering of variable density data is realized. Because the trajectory data are collected by the experiment and they are not sufficient enough, some hot spots are still not effectively reflected. In future studies, with sufficient tourist trajectory data, the data of the staying points in the peak season, off-season or different time periods of the scenic spot can be extracted separately for cluster analyses, and the change of hotspots can be further investigated in different seasons or different time periods. Based on the results of these hotspot analyses, not only the dynamic attraction list can be dynamically adjusted in the smart guide APP, but also the technology such as cloud push can be used to intelligently push the traffic information around the hot spot to the tourists, reminding the tourists to choose the scenic spots reasonably. Moreover, the scenic spot management personnel can also use cluster analyses to assist peak flow and emergency management. The trajectory data in this study only include time and space information. In the future, multiple data mining algorithms should be combined for comprehensive analysis with more attribute data collected such as age, gender, occupation, and hobbies. Additionally, more patterns and features can be explored, and this analysis offers personalized travel push services and intelligent travel management services.

Acknowledgement

This work is partially supported by Hunan Provincial Education Department focused research project (No. 19A283), China.

References

- [1] P. Wang and G. Tu, "Localization algorithm of wireless sensor network based on matrix reconstruction," *Computer Communications*, vol. 154, pp. 216-222, 2020.
- [2] S. Miyazawa, X. Song, T. Xia, R. Shibasaki, and H. Kaneda, "Integrating GPS trajectory and topics from Twitter stream for human mobility estimation," *Frontiers of Computer Science*, vol. 13, no. 3, pp. 460-470, 2019.
- [3] A. Scuttari and G. Isetti, "E-mobility and sustainable tourism transport in remote areas: insights from the Alpine case study of South Tyrol (IT)," *Zeitschrift für Tourismuswissenschaft*, vol. 11, no. 2, pp. 237-256, 2019.
- [4] J. H. Nilsson, "Urban bicycle tourism: Path dependencies and innovation in Greater Copenhagen," *Journal of Sustainable Tourism*, vol. 27, no. 11, pp. 1648-1662, 2019.
- [5] A. Eldridge, "Strangers in the night: nightlife studies and new urban tourism," *Journal of Policy Research in Tourism, Leisure and Events*, vol. 11, no. 3, pp. 422-435, 2019.
- [6] C. Guo, B. Yang, J. Hu, C. S. Jensen, and L. Chen, "Context-aware, preference-based vehicle routing," *The VLDB Journal*, 2020. <https://doi.org/10.1007/s00778-020-00608-7>
- [7] R. Lange, F. Durr, and K. Rothermel, "Efficient real-time trajectory tracking," *The VLDB Journal*, vol. 20, no. 5, pp. 671-694, 2011.
- [8] P. Zhao, X. Liu, J. Shen, and M. Chen, "A network distance and graph-partitioning-based clustering method for improving the accuracy of urban hotspot detection," *Geocarto International*, vol. 34, no. 3, pp. 293-315, 2019.
- [9] C. Liu, Y. Zhou, and X. Li, "Mining semantic trajectory frequent pattern and car pooling application research," *Computer Engineering and Applications*, vol. 55, no. 15, pp. 96-103, 2019.
- [10] Z. Li and K. Luo, "Research on adaptive parameters determination in DBSCAN algorithm," *Computer Engineering and Applications*, vol. 52, no. 3, pp. 70-73, 2016.
- [11] M. Liu, X. Pan, S. Gao, S. Xin, and Y. Zhou, "Segmentation for indoor scenes based on DBSCAN clustering," *Journal of Computer-Aided Design & Computer Graphics*, vol. 71, no. 7, pp. 1183-1193, 2019.
- [12] J. H. Kim, J. H. Choi, K. H. Yoo, and A. Nasridinov, "AA-DBSCAN: an approximate adaptive DBSCAN for finding clusters with varying densities," *The Journal of Supercomputing*, vol. 75, pp. 142-169, 2019.



Yifan Liao <https://orcid.org/0000-0002-7719-5526>

He received his M.S. degree in Circuits and Systems in 2006 from the Hunan Normal University, China. He is currently an Associate Professor in the School of Information & Mechanical Electrical Engineering at Hunan International Economics University. His research interests cover control science, computer graphics, artificial intelligence and pattern recognition.