

Research Article

An Improved Fuzzy Trajectory Clustering Method for Exploring Urban Travel Patterns

Fang Liu ¹, Wei Bi ², Wei Hao ¹, Fan Gao ² and Jinjun Tang ²

¹School of Traffic and Transportation Engineering, Changsha University of Science and Technology, Changsha 410114, China

²Smart Transportation Key Laboratory of Hunan Province, School of Traffic and Transportation Engineering, Central South University, Changsha 410075, China

Correspondence should be addressed to Wei Bi; biweibw@csu.edu.cn

Received 15 December 2020; Revised 14 January 2021; Accepted 25 January 2021; Published 4 February 2021

Academic Editor: Yong Wang

Copyright © 2021 Fang Liu et al. 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.

Exploring urban travel patterns can analyze the mobility regularity of residents to provide guidance for urban traffic planning and emergency decision. Clustering methods have been widely applied to explore the hidden information from large-scale trajectory data on travel patterns exploring. How to implement soft constraints in the clustering method and evaluate the effectiveness quantitatively is still a challenge. In this study, we propose an improved trajectory clustering method based on fuzzy density-based spatial clustering of applications with noise (TC-FDBSCAN) to conduct classification on trajectory data. Firstly, we define the trajectory distance which considers the influence of different attributes and determines the corresponding weight coefficients to measure the similarity among trajectories. Secondly, membership degrees and membership functions are designed in the fuzzy clustering method as the extension of the classical DBSCAN method. Finally, trajectory data collected in Shenzhen city, China, are divided into two types (workdays and weekends) and then implemented in the experiment to explore different travel patterns. Moreover, three indices including Silhouette Coefficient, Davies–Bouldin index, and Calinski–Harabasz index are used to evaluate the effectiveness among the proposed method and other traditional clustering methods. The results also demonstrate the advantage of the proposed method.

1. Introduction

Travel patterns can be explored by analyzing the travel characteristics of moving objects (vehicles and humans), which reflects peoples' travel regularity, traffic congestion regularity, and social activity pattern. Travel patterns have been applied in many areas, for instance, providing the decision information for urban planning and emergency [1–4], analyzing and optimizing the path to provide personalized travel recommendations for residents [5–8], vehicles dispatching [9, 10], and station optimization and selection [11]. These applications can prove insights for urban construction and development.

Nowadays, more and more researchers and scholars explore urban travel patterns using large-scale trajectory data, which contain huge hidden information about travel feature and regularity. Several approaches have been used for

this application. (1) Clustering methods: trajectory clustering has been gaining increasing interest in recent years, and it generally requires two components, similarity measurement and clustering algorithm [12–15]. (2) Spatial statistics: Ni et al. [16] employed a spatial econometric model for travel flow analysis to explore factors that influence travel demand; Zhang et al. [17] proposed a Bayesian hierarchical approach for modeling the destination choice behavior considering the unavailable factors and spatio-temporal correlations; Kamruzzaman et al. [18] estimated the effects of urban form and spatial biases on residential mobility. (3) Deep learning: especially nonparametric deep learning methods have proven to give more accurate predictions in urban traffic forecast [19, 20]. (4) Classical traffic theory models: some classical models have been used in travel patterns analysis. For instance, using combined Markov chain and multinomial logit model [21] to improve the

accuracy of travel destination prediction and using traffic assignment models [22] for O-D matrix estimation.

In the abovementioned methods, the clustering method is an unsupervised learning process that classifies datasets on the basis of similarity. By defining a reasonable similarity criterion, the clustering method can effectively excavate hidden information from massive trajectory data and thus reveal travel patterns. Some researchers adopt traditional clustering methods for travel patterns analysis, such as C-means [23], shared nearest neighbor clustering [24], DBSCAN [25, 26], hierarchical clustering method [27, 28], and optics algorithm [29, 30]. Besides, there are also some researchers who propose several modified clustering methods to obtain better clustering results: a modified bee colony optimization (MBCO) [31] is proposed which introduces the approach based on probability selection; a hybrid model-fused k -means and fuzzy c -means clustering with the modified cluster centroid (FKMFCM-MCC) [32]; Wang et al. [33] presented a modified find density peaks (MFDP) algorithm to transform the high-dimensional points into two-dimensional, and it expressed good potential for application.

The literatures mentioned above have provided practical applications of travel patterns analysis, and several classical or modified clustering methods have been proved that they can achieve good clustering results to explore residents' travel patterns. However, there are also some limitations need to be improved in trajectory clustering for travel patterns analysis. Firstly, as for similarity measurements, lots of literatures only consider the spatial attribute of trajectories, and fewer literatures consider the multiattributes (such as temporal, directional, and other characteristics) of trajectories, ignoring other attributes will cause inaccurate results and unreasonable travel patterns. Besides, many researchers who consider multiattributes do not assign weights to different attributes. The configuration and discussion of weights is necessary because different attributes have different influences on trajectory clustering. Secondly, trajectory data have the attribute of faint borders and overlapping borders; thus, the efficient algorithm which has soft constraints is required to handle such dataset, while lots of researchers consider the ideology of hard divided clustering. Finally, many works pay attention on improving clustering algorithm to reduce computational complexity and obtain better results, and there are lacks in using multiple indices to the quantitative evaluation of results.

Fuzzy clustering methods have the ability to identify clusters with variable density distributions and partially overlapping borders [34]. In fuzzy clustering methods, the framework of the fuzzy set theory is defined with the aim to make up for the shortage of crisp clustering algorithms. Generally, the clustering methods realize soft constraints by introducing weight entropy, membership function, or fuzzy distance function. Currently, the fuzzy clustering methods have been applied in many areas, such as predictive models: Seresht et al. [35] proposed a fuzzy clustering algorithm and assign weights to the fuzzy inference systems to improve the accuracy of predictive models; virus research: Mahmoudi et al. [36] used fuzzy clustering technique to compare the

spread of COVID-19 in many countries; image segmentation: Wu and Chen [37] proposed a novel fuzzy clustering method to improve the robustness of the exiting picture clustering. Moreover, some internal evaluation indices are adopted for trajectory data which are unlabeled in clustering results analysis: silhouette coefficient [38, 39]; Davies–Bouldin index [40]; Calinski–Harabasz index [41]; Q-Measure [42]; Dunn's index [39, 43].

The main contributions of this paper include the following several aspects. Firstly, an improved fuzzy clustering method based on DBSCAN is proposed to cluster taxi trajectory data, in which fuzzy theory is introduced to defining membership functions. It is different from traditional DBSCAN algorithm and can effectively deal with taxi trajectory data which have the attributes of faint borders and overlapping borders. The greatest advantage of the proposed method is introducing the theory of soft constrain, which can divide trajectories into reasonable clusters. While traditional DBSCAN uses hard constrain, when a trajectory may belong to either cluster 1 or cluster 2, the algorithm divides it into first cluster, and this is unreasonable. Secondly, we define the trajectory distance considering the combination of spatial, temporal, and directional attributes to measure the similarity between trajectories. Besides, the weights of different attributes are optimized in trajectory distance function to obtain better results. Finally, several internal evaluation indices are used in model comparison, and the results show the effectiveness and advantage of the proposed method comparing with other classical clustering methods.

The remainder of this paper is organized as follows. Section 2 introduces the framework and theory of the proposed method. In Section 3, data description and case results' analysis of Shenzhen city are introduced. In Section 4, the evaluation among compared approaches is conducted, and the results are then discussed. In Section 5, we summarize the conclusion of this paper.

2. Methodology

2.1. Framework. The proposed method for taxi trajectory clustering in this paper is shown in Figure 1 as a framework. The framework mainly contains three aspects: (1) after preprocessing the initial trajectory data, the trajectory distance can be calculated with the combination of spatial, temporal, and directional distance using the weight coefficients; (2) the TC-FDBSCAN is adopted to cluster trajectories, which need to determine the weight coefficients and other algorithm parameters; (3) three indices are used to evaluate the clustering results.

2.2. Multicharacteristics Similarity Measurement. A trajectory is denoted by TR in this paper. $TR = \{(p_1, t_1), (p_2, t_2), \dots, (p_k, t_k)\}$, where $p_k = (p_k^x, p_k^y)$ denotes the location (abscissa p_k^x and ordinate p_k^y , the longitude and latitude, are converted to plane coordinates) of track point k , and t_k denotes the recording time of track point k . Next, we introduce measurements of multicharacteristics similarity

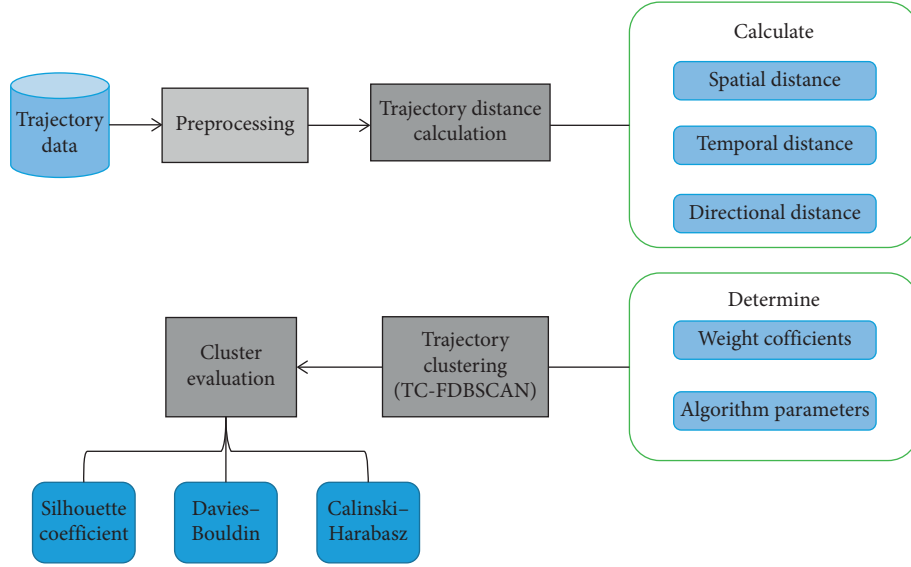


FIGURE 1: Framework of the trajectory clustering method.

between two trajectories TR_A and TR_B , where $TR_A = \{(p_{A1}, t_{A1}), (p_{A2}, t_{A2}), \dots, (p_{Ai}, t_{Ai})\}$ and $TR_B = \{(p_{B1}, t_{B1}), (p_{B2}, t_{B2}), \dots, (p_{Bj}, t_{Bj})\}$.

2.2.1. Spatial Distance between Trajectories. Determining a rule to calculate the spatial distance of track points between trajectories is the key to measure the spatial similarity of trajectories. Hausdorff distance is a common distance measurement method for two points and can be introduced

into trajectory dataset. Figure 2 shows the Hausdorff distance between two trajectories.

Traditional Hausdorff distance calculates the Max-Min distance, which can only measure the dispersion of the distance between trajectories. And, it is susceptible to local shapes of trajectories. In order to improve its robustness to local effects, a modified Hausdorff distance is defined as follows:

$$\begin{aligned}
 h(TR_A, TR_B) &= \frac{1}{n_i} \sum_{p_{Ai} \in TR_A} \left\{ \min_{p_{Bj} \in TR_B} (\text{dist}(p_{Ai}, p_{Bj})) \right\}, \\
 h(TR_B, TR_A) &= \frac{1}{n_j} \sum_{p_{Bj} \in TR_B} \left\{ \min_{p_{Ai} \in TR_A} (\text{dist}(p_{Bj}, p_{Ai})) \right\}, \\
 H(TR_A, TR_B) &= \max(h(TR_A, TR_B), h(TR_B, TR_A)),
 \end{aligned} \tag{1}$$

where n_i denotes the number of track points in TR_A , n_j denotes the number of points in TR_B , $\text{dist}(p_{Ai}, p_{Bj})$ is the Euclidean distance between points p_{Ai} and p_{Bj} , and $H(TR_A, TR_B)$ is an absolute Hausdorff distance, which denotes the spatial distance between two trajectories.

2.2.2. Temporal Distance between Trajectories. The trajectory is represented as an interval, which is related to time of starting points and time of ending points. Considering the influence of trajectory duration, the temporal distance between two trajectories is defined as follows:

$$\text{Tem}(TR_A, TR_B) = \begin{cases} 1, & T_A \cap T_B = \emptyset, \\ \frac{|t_{A1} - t_{B1}| + |t_{Ai} - t_{Bj}|}{(t_{Ai} - t_{A1}) + (t_{Bj} - t_{B1})}, & T_A \cap T_B \neq \emptyset, \end{cases} \tag{2}$$

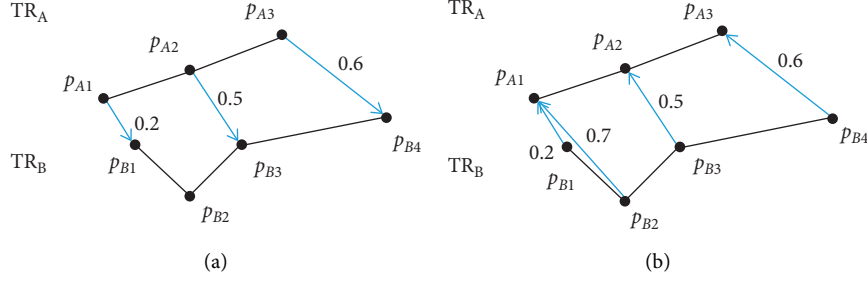


FIGURE 2: Hausdorff distance between two trajectories.

where $T_A = \{t_{A1}, t_{A2}, \dots, t_{Ai}\}$ and $T_B = \{t_{B1}, t_{B2}, \dots, t_{Bj}\}$.

Only if the starting time and ending time of two trajectories are the same ($t_{A1} = t_{B1}$, $t_{Ai} = t_{Bj}$), the temporal distance is 0, which means two trajectories are completely similar in temporal characteristic. When two trajectories are separated in temporal characteristic, they are not completely similar and have a temporal distance $\text{Tem}(\text{TR}_A, \text{TR}_B)$ between 0 and 1.

2.2.3. Directional Distance between Trajectories. Linear directional mean is commonly used to describe the trend or average direction of a set of lines. On the basis of linear directional mean, for trajectory TR_A , treating each trajectory segment (which consists of two consecutive points in TR_A) as a line, its average direction is defined as follows:

$$\gamma_{\text{TR}_A} = \arctan \left| \frac{\sum_{k=1}^{n_i-1} \sin \theta'_{Ak}}{\sum_{k=1}^{n_i-1} \cos \theta'_{Ak}} \right|,$$

$$\gamma'_{\text{TR}_A} = \begin{cases} \gamma_{\text{TR}_A}, & \sum_{k=1}^{n_i-1} \cos \theta'_{Ak} > 0, \sum_{k=1}^{n_i-1} \sin \theta'_{Ak} \geq 0, \\ 180^\circ - \gamma_{\text{TR}_A}, & \sum_{k=1}^{n_i-1} \cos \theta'_{Ak} < 0, \sum_{k=1}^{n_i-1} \sin \theta'_{Ak} \geq 0, \\ 180^\circ + \gamma_{\text{TR}_A}, & \sum_{k=1}^{n_i-1} \cos \theta'_{Ak} < 0, \sum_{k=1}^{n_i-1} \sin \theta'_{Ak} \leq 0, \\ 360^\circ - \gamma_{\text{TR}_A}, & \sum_{k=1}^{n_i-1} \cos \theta'_{Ak} > 0, \sum_{k=1}^{n_i-1} \sin \theta'_{Ak} \leq 0, \end{cases}$$

$$\theta_{Ak} = \arctan \left| \frac{p_{Ak+1}^y - p_{Ak}^y}{p_{Ak+1}^x - p_{Ak}^x} \right|,$$

$$\theta'_{Ak} = \begin{cases} \theta_{Ak}, & p_{Ak+1}^x - p_{Ak}^x > 0, p_{Ak+1}^y - p_{Ak}^y \geq 0, \\ 180^\circ - \theta_{Ak}, & p_{Ak+1}^x - p_{Ak}^x < 0, p_{Ak+1}^y - p_{Ak}^y \geq 0, \\ 180^\circ + \theta_{Ak}, & p_{Ak+1}^x - p_{Ak}^x < 0, p_{Ak+1}^y - p_{Ak}^y \leq 0, \\ 360^\circ - \theta_{Ak}, & p_{Ak+1}^x - p_{Ak}^x > 0, p_{Ak+1}^y - p_{Ak}^y \leq 0, \end{cases} \quad (3)$$

where θ'_{Ak} is the real direction of trajectory segment $\{(p_{Ak}, t_{Ak}), (p_{Ak+1}, t_{Ak+1})\}$, which represents the angle rotated counterclockwise due east, $\theta'_{Ak} \in [0^\circ, 360^\circ]$.

The angle of linear directional mean between two trajectories is defined as follows:

$$\gamma(\text{TR}_A, \text{TR}_B) = |\gamma'_{\text{TR}_A} - \gamma'_{\text{TR}_B}|. \quad (4)$$

2.2.4. Trajectory Distance. Firstly, the spatial and directional distance need to be Min-Max normalized and converted to the dimensionless value, while the temporal distance does not need to be normalized because its value is between 0 and 1; then, the corresponding distance can be defined as follows:

$$\text{SDist}(\text{TR}_A, \text{TR}_B) = \frac{H(\text{TR}_A, \text{TR}_B) - \min\{H\}}{\max\{H\} - \min\{H\}},$$

$$\text{TDist}(\text{TR}_A, \text{TR}_B) = \text{Tem}(\text{TR}_A, \text{TR}_B), \quad (5)$$

$$\text{DDist}(\text{TR}_A, \text{TR}_B) = \frac{\gamma(\text{TR}_A, \text{TR}_B) - \min\{\gamma\}}{\max\{\gamma\} - \min\{\gamma\}},$$

where H and γ are the set of spatial and directional distance between all pairs of trajectories, respectively.

In combination with spatial, temporal, and directional distance, the trajectory distance is defined considering the influence of weights:

$$\begin{aligned} \text{TRDist}(\text{TR}_A, \text{TR}_B) &= \alpha \cdot \text{SDist}(\text{TR}_A, \text{TR}_B) \\ &\quad + \beta \cdot \text{TDist}(\text{TR}_A, \text{TR}_B) \\ &\quad + \omega \cdot \text{DDist}(\text{TR}_A, \text{TR}_B), \\ \alpha + \beta + \omega &= 1, \quad \alpha \geq 0, \beta \geq 0, \omega \geq 0, \end{aligned} \quad (6)$$

where α , β , and ω are weight coefficients.

2.3. Trajectory Clustering Method TC-FDBSCAN. In this section, based on the classical DBSCAN method, an improved TC-FDBSCAN (trajectory clustering method based on fuzzy density-based spatial clustering of applications with noise) method is proposed in trajectory data clustering. In this method, membership functions are introduced to achieve fuzziness and parameters MinTRs (minimum trajectories in neighborhood) and ε (radius of neighborhood)

in classical DBSCAN are replaced by MinTRs_{\min} (minimum value of minimum trajectories in neighborhood), MinTRs_{\max} (maximum value of minimum trajectories in neighborhood), ϵ_{\min} (minimum radius), and ϵ_{\max} (maximum radius).

Several extended definitions of the modified method are described in detail as follows.

Definition 1. Neighborhood: the region within a definite radius of a trajectory.

Definition 2. Core trajectory: the number of trajectories in the neighborhood of a trajectory is greater than a definite value.

$$\mu_{\epsilon}(\text{TR}, \text{TR}_l) = \begin{cases} 1, & |\text{TRDist}(\text{TR}, \text{TR}_l)| \leq \epsilon_{\min}, \\ \frac{\epsilon_{\max} - |\text{TRDist}(\text{TR}, \text{TR}_l)|}{\epsilon_{\max} - \epsilon_{\min}}, & \epsilon_{\min} < |\text{TRDist}(\text{TR}, \text{TR}_l)| < \epsilon_{\max}, \\ 0, & |\text{TRDist}(\text{TR}, \text{TR}_l)| > \epsilon_{\max}. \end{cases} \quad (8)$$

This membership function considers the fuzzies of neighborhood, which causes a trajectory having a specified membership degree in the fuzzy neighborhood of another trajectory.

$$\mu_{\text{MinTRs}}(\text{den}(\text{TR})) = \begin{cases} 1, & \text{den}(\text{TR}) \geq \text{MinTRs}_{\max}, \\ \frac{\text{den}(\text{TR}) - \text{MinTRs}_{\min}}{\text{MinTRs}_{\max} - \text{MinTRs}_{\min}}, & \text{MinTRs}_{\min} < \text{den}(\text{TR}) < \text{MinTRs}_{\max}, \\ 0, & \text{den}(\text{TR}) \leq \text{MinTRs}_{\min}. \end{cases} \quad (9)$$

This membership function considers the fuzzies of core trajectory, which causes the number of trajectories in the fuzzy neighborhood having a specified membership degree.

Definition 6. Core membership degree: if $\mu_{\text{MinTRs}}(\text{den}(\text{TR})) > 0$, then trajectory TR is a fuzzy core trajectory, and it

Definition 3. The local density of a trajectory:

$$\text{den}(\text{TR}) = \sum_{\text{TR}_l \in \text{neighbor}(\text{TR}, \epsilon_{\max})} \mu_{\epsilon}(\text{TR}, \text{TR}_l), \quad (7)$$

where $\text{neighbor}(\text{TR}, \epsilon_{\max}) = \{\text{TR}_l, |\text{TRDist}(\text{TR}, \text{TR}_l)| < \epsilon_{\max}\}$ and $\text{den}(\text{TR})$ denotes the number of trajectories in neighborhood with membership degree $\mu_{\epsilon}(\text{TR}, \text{TR}_l)$.

Definition 4. ϵ membership function:

Definition 5. MinTRs membership function:

belongs to a cluster with core membership degree $\text{fuzzycore}(\text{TR}) = \mu_{\text{MinTRs}}(\text{den}(\text{TR}))$.

Definition 7. Border membership degree: if $\mu_{\text{MinTRs}}(\text{den}(\text{TR})) = 0$, then trajectory TR should be a border or noise trajectory. Then, TR can be a fuzzy border trajectory which belongs to a cluster with border membership degree:

$$\text{fuzzyborder}(\text{TR}) = \min_{\text{TR}_l \in \text{neighborcore}(\text{TR})} (\min(\mu_{\text{MinTRs}}(\text{den}(\text{TR}_l)), \mu_{\epsilon}(\text{TR}, \text{TR}_l))), \quad (10)$$

where $\text{neighborcore}(\text{TR}) = \{\text{TR}_l, \mu_{\text{MinTRs}}(\text{den}(\text{TR}_l)) > 0 \text{ and } \mu_{\epsilon}(\text{TR}, \text{TR}_l) > 0\}$.

The trajectory will be a noise trajectory if it is not a fuzzy core trajectory or fuzzy border trajectory.

Definition 8. Directly density-reachable: if trajectory TR_j is in the fuzzy neighborhood of trajectory TR_i and TR_i is a fuzzy core trajectory, then TR_j is directly density-reachable to TR_i .

Definition 9. Density-reachable: if each TR_n is directly density-reachable to TR_{n-1} under the condition of TR_n and TR_{n-1} are in the $\{TR_i, \dots, TR_{n-1}, TR_n, \dots, TR_j\}$, then TR_j is density-reachable to TR_i .

Definition 10. Density-connected: if TR_k is density-reachable to both TR_i and TR_j , then TR_i and TR_j are density-connected.

The aim of the TC-FDBSCAN is to divide the regions with high density into clusters, which are the largest sets of density-connected trajectories. The major difference from classical DBSCAN is that a trajectory is determined to be a core trajectory or border trajectory considering membership degree, which means the possibility of this trajectory to belong to clusters.

The process is demonstrated as follows which is similar with classical DBSCAN:

- (1) Randomly select a trajectory to visit. If it is a fuzzy core trajectory, then add it into a new cluster with membership degree computed by equation (11) and add other trajectories in its fuzzy neighborhood into an alternate set.
- (2) Visit other trajectories in the alternate set; if the trajectories satisfy the fuzzy core trajectory condition, then add them into the original cluster with membership degree computed by equation (11) and add the trajectories in their fuzzy neighborhood; if they do not satisfy the fuzzy core trajectory condition, then add them into original cluster as fuzzy border trajectories with membership degree computed by equation (12).
- (3) Repeat step (2) until all trajectories in the alternate set have been visited.
- (4) Repeat step (1) to step (3) until all trajectories in the dataset have been visited.

There is a simple approach to allow users to determine two percentages, per_{\min} and per_{\max} . And, then, use $\text{per}_{\min} \cdot \max\{\text{TRDist}\}$ and $\text{per}_{\max} \cdot \max\{\text{TRDist}\}$ to determine the values of ε_{\min} and ε_{\max} , where $\max\{\text{TRDist}\}$ denotes the

maximum distance between all pairs of trajectories. For MinTRs_{\min} and MinTRs_{\max} , a curve can be drawn, in which the x -coordinate is ε and y -coordinate is the number of trajectories in the set where the distance between each other is equal to ε . This curve is not monotonically decreasing; then, the corresponding y -coordinate values in the first two bends can be selected as MinTRs_{\min} and MinTRs_{\max} .

2.4. Cluster Evaluation Indices. In the relevant studies of cluster evaluation, as the experimental data is labeled and which cluster the sample belongs to is known in advance, so the method based on accuracy such as Purity, Rand Index, and Accuracy can be used. While the trajectory data is unlabeled, there is no external information that can be used to verify the authenticity of the clustering results. Then, several internal evaluation indices which consider the geometric structure of data are required to evaluate the effect of clustering results.

Silhouette coefficient (SC) [36]:

$$SC = \frac{1}{n_C} \sum_{i=1}^{n_C} \frac{b_i - a_i}{\max(a_i, b_i)}, \quad (11)$$

$$b_i = \min_{h=1,2,\dots,n_C, h \neq i} \left(\frac{1}{n_{C_h}} \sum_{TR_h \in C_h} \text{TRDist}(TR_i, TR_{C_h}) \right), \quad (12)$$

$$a_i = \frac{1}{n_{C_i}} \left(\sum_{TR_i \in C_i} \text{TRDist}(TR_i, TR_{C_i}) \right), \quad (13)$$

where n_C is the number of clusters, C_i denotes the i th cluster, n_{C_i} denotes the number of trajectories in C_i , and TR_{C_i} denotes the center of C_i . The larger value of SC indicates better clustering results.

Davies–Bouldin index (DB) [44]:

$$DB = \frac{1}{n_C} \sum_{i=1}^{n_C} \max_{h=1,2,\dots,n_C, i \neq h} \frac{(1/n_{C_i})(\sum_{TR_i \in C_i} \text{TRDist}(TR_i, TR_{C_i})) + (1/n_{C_h})(\sum_{TR_h \in C_h} \text{TRDist}(TR_h, TR_{C_h}))}{\text{TRDist}(TR_{C_i}, TR_{C_h})}, \quad (14)$$

where C_h denotes the h th cluster, n_{C_h} denotes the number of trajectories in C_h , and TR_{C_h} denotes the center of C_h . The smaller value of DB indicates better clustering results.

Calinski–Harabasz index (CH) [45]:

$$CH = \sum_{i=1}^{n_C} \left(\frac{n_{C_i} \text{TRDist}^2(TR_{C_i}, TR_{C_i}) / (n_C - 1)}{\sum_{i=1}^{n_C} \sum_{TR_i \in C_i} (\text{TRDist}^2(TR_i, TR_{C_i}) / (n_{TR} - n_C))} \right), \quad (15)$$

where n_{TR} denotes the number of all trajectories in dataset and TR_C denotes the center of dataset. The larger value of CH indicates better clustering results.

3. Case Study

3.1. Data Description. The research area of the experiments is Shenzhen City, China. Shenzhen city consists of eight administrative districts (Futian, Luohu, Nanshan, Yantian,

Baoan, Longgang, Longhua, Pingshan, and Guangming) and one functional district (Dapeng), which are located between 113°46' E to 114°37' E and 22°27' N to 22°52' N. In this study, taxi trajectory data is used in experiment, and it is composed of a series of sample points collected by vehicular GPS equipment.

Each sample point includes the information of the license plate, location (latitude and longitude), recording time, and instantaneous speed and state (0 represents vacant and 1 represents occupied). We use the data collected by 1000 taxis during one week in May, 2019, and we divide them into two groups to conduct experiment separately, which are weekdays (from May 13th to May 17th) and weekends (from May 18th to May 19th). In order to reduce the influence of abnormal points on clustering results, the original data is preprocessed firstly, which mainly includes eliminating the outliers (latitude and longitude are 0 or out of right range and the state is neither 0 or 1) and interpolating the missing points (recording time interval discontinuity). Finally, 2,499,880 original points during the workdays generate 82,144 passenger-occupied trajectories (49,456 trajectories are valid), and 107,186 original points during the weekends generate 34,384 passenger-occupied trajectories (22,774 trajectories are valid).

The travel time and travel distance of trajectories on workdays and weekends can be calculated, respectively, as shown in Figure 3. The percentage represents the ratio of the number of trajectories in the interval to all trajectories. It can be concluded that both on workdays and weekends, most passengers prefer to take taxi in a short (0 to 5 km) or medium (5 to 15 km) travel distance and takes no more than 20 minutes. Few passengers take more than 1 hour by taxi.

3.2. Parameter Configuration. In the proposed method, the parameters including α , β , ω , ϵ_{\min} , ϵ_{\max} , MinTRs_{\min} , and MinTRs_{\max} are needed to be determined to obtain reasonable clustering results. In parameter configuration and comparison, we adjust parameters and select the reasonable parameter combination according to the Silhouette Coefficient (SC) of the clustering results. Firstly, set up several combinations of weight coefficients to compare the clustering results, and the trajectory distance TRDist can be calculated meanwhile. Secondly, determine two percentages $\text{per}_{\min} = 40\%$ and $\text{per}_{\max} = 60\%$ according to the approach mentioned in Section 2.3; then, ϵ_{\min} and ϵ_{\max} can be determined according to the value of $\max\{\text{TRDist}\}$. In this part, $\max\{\text{TRDist}\}$ is equal to 1 in these combinations, so we select $\epsilon_{\min} = 0.4$ and $\epsilon_{\max} = 0.6$. Thirdly, the statistical curve is drawn to select MinTRs_{\min} and MinTRs_{\max} as mentioned in Section 2.3. For workday data, $\text{MinTRs}_{\min} = 1876$ and $\text{MinTRs}_{\max} = 2353$, which are shown in Figure 4(b). While, for weekend data, $\text{MinTRs}_{\min} = 782$ and $\text{MinTRs}_{\max} = 986$, which are shown in Figure 5(b). Finally, the optimal parameter combination is determined in the condition that the value of SC is largest in all combinations.

Next, the process of parameter configuration for workday data and weekend data are demonstrated as follows, respectively. Totally, we compare twenty combinations

of weight coefficients, and the detailed results including the SC, the number of clusters, and the number of noise trajectories of each clustering result are shown in Tables 1 and 2. We can observe the points which have largest value of SC from Figures 4(a) and 5(a). These points reflect that the clustering results under the corresponding weight combination are the best result for the case study. And, the best results are highlighted in bold in Tables 1 and 2.

Finally, for workday data, we select $\alpha = 0.5$, $\beta = 0.3$, $\omega = 0.2$, $\epsilon_{\min} = 0.4$, $\epsilon_{\max} = 0.6$, $\text{MinTRs}_{\min} = 1876$, and $\text{MinTRs}_{\max} = 2353$, while, for weekend data, we select $\alpha = 0.4$, $\beta = 0.4$, $\omega = 0.2$, $\epsilon_{\min} = 0.4$, $\epsilon_{\max} = 0.6$, $\text{MinTRs}_{\min} = 782$, and $\text{MinTRs}_{\max} = 986$.

3.3. Clustering Results. Based on the selected parameters, we then adopt the proposed method to cluster taxi trajectory data in Shenzhen city. Figure 6 shows travel patterns of taxi trajectory data collected on workday and weekend. Tables 3 and 4 show the detailed temporal and directional information of each cluster. The value of main direction represents the angle of rotation counterclockwise in terms of east.

We can conclude some findings comparing the final results between workday and weekend data. Several common phenomena are observed as follows. (1) Most of the trajectories in final clusters are located in arterial roads of city and are concentrated in the downtown area. (2) Some main commuter roads are both detected in clusters (for example, Beijing-Hong Kong-Macau expressway and Binhe avenue are two main roads where most trajectories of clusters located in, which can be observed in Figure 6(a) Cluster 5 and Figure 6(b) Cluster 1). (3) Some clusters do not show obvious characteristics on temporal attribute, while they are prominent on spatial and directional attributes (for example, the main travel time periods of Cluster 4 and Cluster 5 in Figure 6(a) do not have a concentrate pattern, while Cluster 4 concentrates on Guangming district, the travel trend is the southeast, and Cluster 5 concentrates on Futian district, the travel trend is towards the east). (4) Clusters which have opposite direction are found in the workday and weekend results, respectively (for example, in Figure 6(a) and Table 3, Cluster 3 and Cluster 4 have opposite direction, since the main direction of Cluster 3 is 138.26° and the main direction of Cluster 4 is 319.35°). It shows that the proposed method can effectively cluster trajectories which have different trends of directions.

There are also obvious differences between workday and weekend. Firstly, the clustering results of weekend data is better than workday data according to the value of SC (workday: 0.6382 and weekend: 0.6418). Secondly, on weekend, the range of clusters are more widespread, and residents tend to travel to the suburb (for example, in Figure 6(b), Cluster 2 and Cluster 3 reflect the larger travel range, the trajectories travel on roads including Pingshan avenue and Pingkui road are added to the final cluster). Thirdly, in contrast to workday, the trajectories of clusters reflect travel in night on weekend (for example, in Figure 6(b) and Table 4, the trajectories in Cluster 4 concentrate on night periods, which during 22:00 to 0:30).

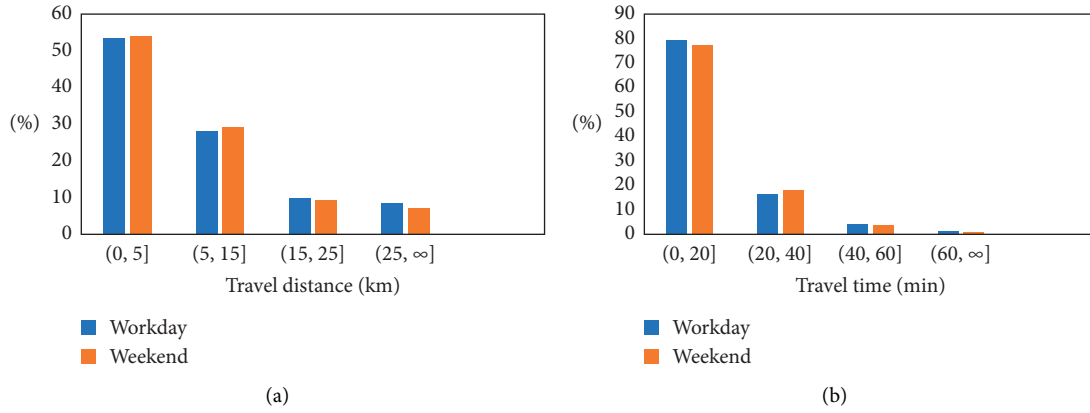


FIGURE 3: The statistical characteristics of trajectories on workdays and weekends: (a) the proportion of trajectories in different travel distance intervals and (b) the proportion of trajectories in different travel time intervals.

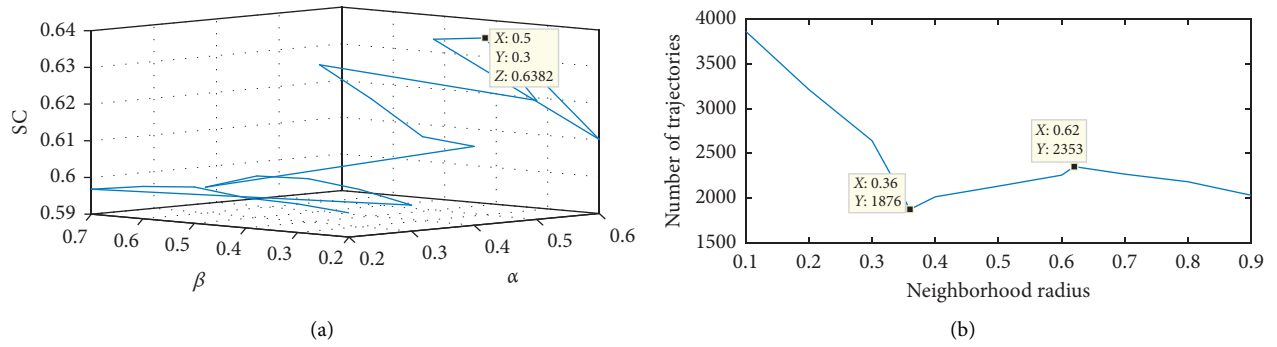


FIGURE 4: Parameter configuration for the data collected on workday: (a) the SC under different combinations of weight coefficients and (b) the curve used to determine MinTRs_{\min} and MinTRs_{\max} .

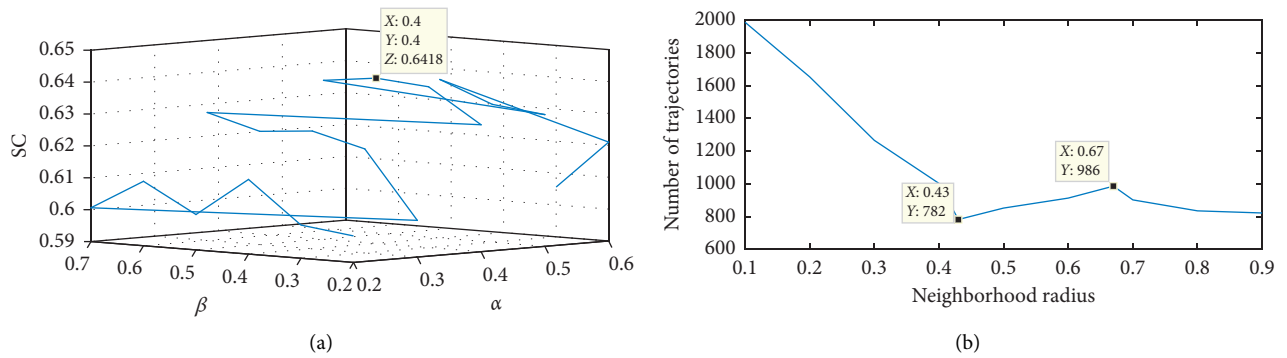


FIGURE 5: Parameter configuration for the data collected on weekend: (a) the SC under different combinations of weight coefficients and (b) the curve used to determine MinTRs_{\min} and MinTRs_{\max} .

TABLE 1: Comparison of different parameter combinations for workday data.

α	β	ω	SC	The number of clusters	The number of noise trajectories
0.2	0.2	0.6	0.5966	4	16351
0.2	0.3	0.5	0.5978	4	15970
0.2	0.4	0.4	0.5982	5	11369
0.2	0.5	0.3	0.5999	5	12158
0.2	0.6	0.2	0.5988	5	14223
0.2	0.7	0.1	0.5968	5	10196
0.3	0.2	0.5	0.5971	5	9945

TABLE 1: Continued.

α	β	ω	SC	The number of clusters	The number of noise trajectories
0.3	0.3	0.4	0.6001	5	12454
0.3	0.4	0.3	0.6018	5	13112
0.3	0.5	0.2	0.6013	5	10898
0.3	0.6	0.1	0.5970	4	15668
0.4	0.2	0.4	0.6115	4	16011
0.4	0.3	0.3	0.6129	5	12447
0.4	0.4	0.2	0.6220	5	12098
0.4	0.5	0.1	0.6300	6	10256
0.5	0.2	0.3	0.6224	6	9963
0.5	0.3	0.2	0.6382	6	10891
0.5	0.4	0.1	0.6366	5	11025
0.6	0.2	0.2	0.6102	7	8861
0.6	0.3	0.1	0.6233	5	10258

TABLE 2: Comparison of different parameter combinations for weekend data.

α	β	ω	SC	The number of clusters	The number of noise trajectories
0.2	0.2	0.6	0.5983	6	5999
0.2	0.3	0.5	0.6004	6	5766
0.2	0.4	0.4	0.6134	6	5988
0.2	0.5	0.3	0.6011	8	5365
0.2	0.6	0.2	0.6102	7	5531
0.2	0.7	0.1	0.6006	7	5688
0.3	0.2	0.5	0.6015	5	6012
0.3	0.3	0.4	0.6226	7	5764
0.3	0.4	0.3	0.6269	7	5841
0.3	0.5	0.2	0.6255	8	5499
0.3	0.6	0.1	0.6301	8	5501
0.4	0.2	0.4	0.6298	7	5223
0.4	0.3	0.3	0.6404	9	4913
0.4	0.4	0.2	0.6418	8	5423
0.4	0.5	0.1	0.6398	9	4826
0.5	0.2	0.3	0.6313	9	5012
0.5	0.3	0.2	0.6332	10	4788
0.5	0.4	0.1	0.6397	8	4978
0.6	0.2	0.2	0.6210	8	4814
0.6	0.3	0.1	0.6356	7	5112

Finally, the number of clusters on weekend is more than weekend and the direction of clusters are more various on weekend which are shown in Figures 6(a) and 6(b).

4. Discussion

In order to verify the effectiveness of the proposed method, we compare it with other commonly used clustering methods in this section. The case data is 10,000 trajectories randomly selected from the original data (from May 13th to May 19th). Three internal evaluation indices such as Silhouette Coefficient, Davies–Bouldin index, and Calinski–Harabasz index as mentioned in Section 2.4 are adopted to evaluate the clustering results.

4.1. Comparative Approaches

4.1.1. Hard C-Means (HCM). HCM or K-means clustering algorithm is an iterative solution of the cluster analysis algorithm. In this method, the data is divided into C groups, which

are randomly selected from C objects as the initial clustering centers. Then, for each object, calculate the distance between every seed clustering center to each object from its nearest cluster center. The clustering centers and the objects assigned to them represent a cluster. Since each sample has been assigned, the clustering center of the cluster is recalculated according to the existing objects in the cluster. This process is repeated until a termination condition is met. The termination condition can be that no (or minimum number) objects are reassigned to different clusters, no (or minimum number) cluster centers change again, and the error sum of squares is locally minimum.

4.1.2. Fuzzy C-Means (FCM). Fuzzy C-means clustering is a kind of clustering algorithm which uses membership degree to determine the degree of each object belonging to a certain cluster. It is an improvement of the earlier HCM clustering method. FCM divides the original data into C fuzzy groups and finds the clustering center of each group in order to minimize the value function of the nonsimilarity index. The main difference between FCM and HCM is FCM applying

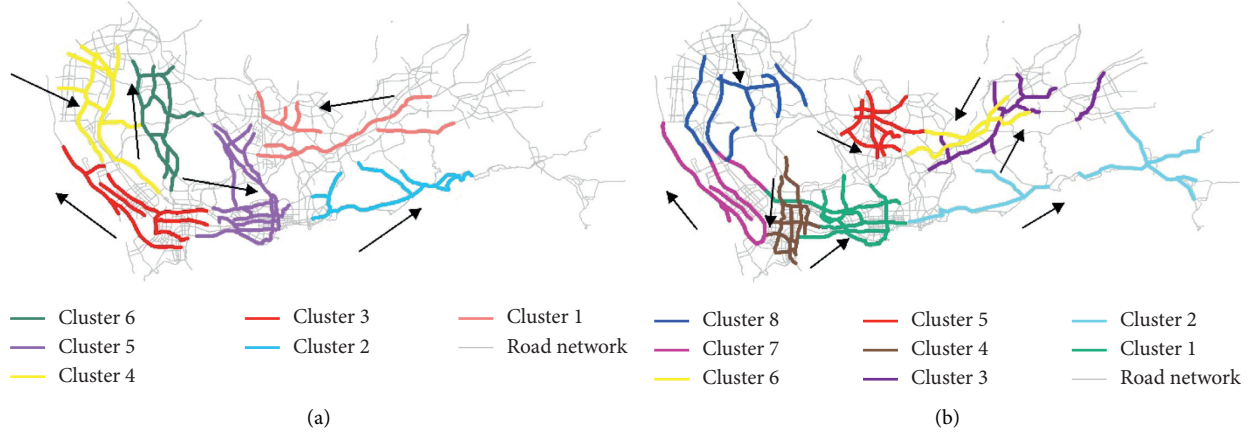


FIGURE 6: Clustering results: (a) workday and (b) weekend.

TABLE 3: Detailed information of each cluster on workday.

Cluster	Main travel time periods	Main direction (angle)
Cluster 1	7:00 to 9:00	189.29°
Cluster 2	16:30 to 19:00	44.38°
Cluster 3	All day	138.26°
Cluster 4	All day	319.35°
Cluster 5	All day	332.17°
Cluster 6	7:00 to 9:00; 16:00 to 20:00	94.32°

TABLE 4: Detailed information of each cluster on weekend.

Cluster	Main travel time periods	Main direction (angle)
Cluster 1	All day	46.18°
Cluster 2	All day	42.06°
Cluster 3	9:00 to 11:00	57.92°
Cluster 4	22:00 to 0:30	264.32°
Cluster 5	20:00 to 22:30	331.23°
Cluster 6	16:00 to 19:00	227.36°
Cluster 7	All day	124.06°
Cluster 8	All day	274.34°

fuzzy division so that each given trajectory is determined by the membership degree between 0 and 1. In accordance with the introduction of fuzzy division, the membership matrix has normalization provisions so that the sum of membership degrees of a dataset is always equal to 1.

4.1.3. Agglomerative Nesting Algorithm (AGNES). Agglomerative nesting algorithm adopts a bottom-up strategy. Each object is initially treated as a cluster, and these clusters are then merged step by step according to some criteria. The distance between two clusters can be determined by the similarity of the closest data in the two different clusters. The merging process of clustering is repeated until all objects meet the number of clusters.

4.1.4. DBSCAN. DBSCAN is a density-based clustering algorithm, which generally assumes that clusters can be determined by the density of the sample distribution. Samples of the same cluster are closely related to each other, and there must be

other samples of the same cluster not far away from any sample of the cluster. A cluster is obtained by grouping closely related samples together. The final result of all clusters is obtained by dividing all groups of closely related samples into different clusters. The definitions of DBSCAN algorithm are similar to the descriptions in Section 2.3 without membership constraint.

4.1.5. Shared Nearest Neighbor Clustering (SNNC). The shared nearest neighbor clustering algorithm was proposed by Jarvis and Patrick, where a link is created between a pair of points p and q , if and only if p and q have each other in their closest k -nearest neighbor [46]. This algorithm is an extension of the DBSCAN. The basic idea of SNNC is based on determining the core points around which clusters with various sizes and shapes are built, without worrying about determining their number [47]. Counting the number of points shared between two points p and q in their k -nearest neighbor list based on the distance metric allows us to determine the similarity between them. The greater the number of shared points, the higher the similarity between p and q .

4.2. Results' Discussion. In order to apply these approaches to the case study in comparison analysis, the distance in HCM, FCM, AGNES, DBSCAN, and SNNC is calculated by the trajectory distance as mentioned in our method. The weight coefficients are determined, so the trajectory distance is described by $TRDist = 0.4 \times SDist(TR_A, TR_B) + 0.3 \times TDist(TR_A, TR_B) + 0.3 \times DDist(TR_A, TR_B)$. The detailed results are shown in Table 5.

Table 5 and Figure 7 show the effectiveness evaluation of different clustering approaches, and we can observe their

TABLE 5: Comparison of different clustering approaches for case data.

Serial number	Approach	Main parameters	SC
1	HCM	The number of clusters: 8	0.563
2	FCM	The number of clusters: 8 Fuzzy degree parameter: 2	0.591
3	AGNES	The number of clusters: 8 Linkage distance: 0.45	0.545
4	DBSCAN	ϵ : 0.45; MinTRs: 350	0.620
5	SNNC	\	0.633
6	TC-FDBSCAN	MinTRs _{min} : 300; MinTRs _{max} : 500; ϵ_{min} : 0.36; ϵ_{max} : 0.54	0.652

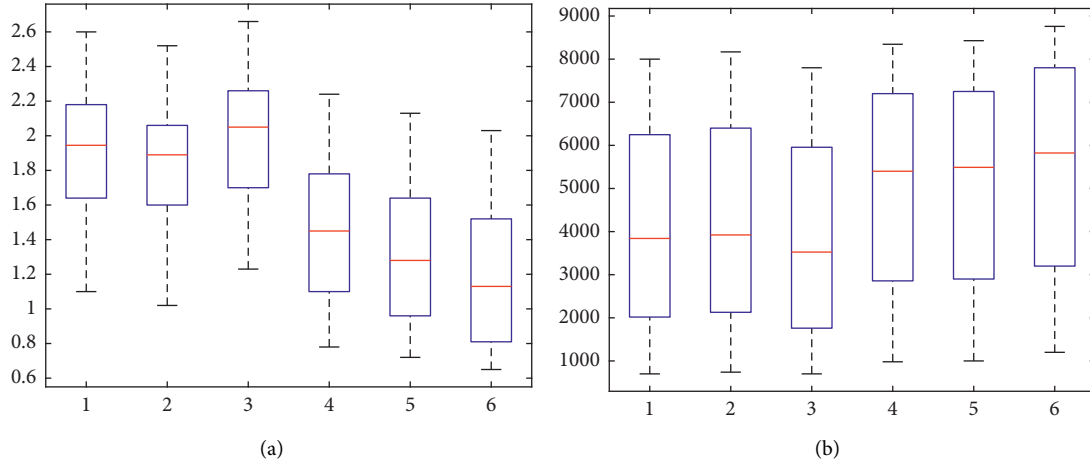


FIGURE 7: Boxplot diagram of the clustering evaluation for case data. (a) DB index. (b) CH index.

ability in analyzing unlabeled data samples. The values of SC show that the proposed TC-FDBSCAN model produces better clustering results than other methods, while AGNES produces worse clustering results than others. In Figure 7, the DB index and CH index describe the similar effect: (1) TC-FDBSCAN is obviously superior to other methods; (2) AGNES is obviously inferior to others; (3) HCM and FCM show the similar results, while FCM is better than HCM; (4) DBSCAN and SNNC show that their clustering performance are close, while SNNC is better than DBSCAN. Overall, the proposed method can find better clustering division and provide high clustering accuracy for large-scale trajectory data in travel pattern analysis.

5. Conclusion

Clustering taxi trajectory based on similarity measurement is a widely applied way to explore urban travel patterns. This study proposes an improved TC-FDBSCAN to uncover urban travel patterns. The taxi trajectory data collected in Shenzhen city is used to evaluate clustering results in the case study. The dataset is divided into two parts, workdays and weekends, which are be used in clustering analysis and model comparison. Some main findings are concluded in following aspects. (1) Both on workdays and weekends, the trajectories in clusters are mainly distributed on the arterial roads. However, clustering results show that, on weekends, the range of residents' travel is wider than that analyzed on

workdays. (2) Introducing the fuzzy theory into traditional DBSCAN algorithm can improve clustering performance according to three evaluation indicators. (3) Different attributes of trajectories have different influences on clustering results according to the values of weight coefficients.

There are still some limitations which need to be improved in future study. On the one hand, other fuzzy clustering methods need to be studied to reduce computational complexity of the algorithm. On the other hand, other fuzzy theory such as weight entropy should be combined in the trajectory clustering method. Moreover, the proposed method is also necessary to be applied in different cities to prove its universality.

Data Availability

The trajectory data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was funded in part by the Innovation-Driven Project of Central South University (no. 2020CX041),

Natural Science Foundation of Hunan Province (no. 2020JJ4752), National Key R&D Program of China (no. 2020YFB1600400), and Foundation of Central South University (no. 502045002).

References

- [1] Y. Yan, S. Zhang, J. Tang, and X. Wang, "Understanding characteristics in multivariate traffic flow time series from complex network structure," *Physica A: Statistical Mechanics and its Applications*, vol. 477, pp. 149–160, 2017.
- [2] R. A. Acheampong, "Spatial structure, intra-urban commuting patterns and travel mode choice: analyses of relationships in the Kumasi Metropolis, Ghana," *Cities*, vol. 96, Article ID 102432, 2020.
- [3] J. J. Tang, J. Hu, W. Hao, X. Q. Chen, and Y. Qi, "Markov chains based route travel time estimation considering link spatio-temporal correlation," *Physica A*, vol. 545, pp. 1–15, 2020.
- [4] Y. Wang, S. Peng, X. Zhou, M. Mahmoudi, and L. Zhen, "Green logistics location-routing problem with eco-packages," *Transportation Research Part E: Logistics and Transportation Review*, vol. 143, Article ID 102118, 2020.
- [5] Y. Wang, X. Ma, Z. Li, Y. Liu, M. Xu, and Y. Wang, "Profit distribution in collaborative multiple centers vehicle routing problem," *Journal of Cleaner Production*, vol. 144, pp. 203–219, 2017.
- [6] J. Tang, S. Zhang, X. Chen, F. Liu, and Y. Zou, "Taxi trips distribution modeling based on Entropy-Maximizing theory: a case study in Harbin city-China," *Physica A: Statistical Mechanics and its Applications*, vol. 493, pp. 430–443, 2018.
- [7] X. Jiao, Y. Xiao, W. Zheng, H. Wang, and C.-H. Hsu, "A novel next new point-of-interest recommendation system based on simulated user travel decision-making process," *Future Generation Computer Systems*, vol. 100, pp. 982–993, 2019.
- [8] Y. Wang, Y. Yuan, X. Guan et al., "Collaborative two-echelon multicenter vehicle routing optimization based on state-space-time network representation," *Journal of Cleaner Production*, vol. 258, Article ID 120590, 2020.
- [9] L. Dimitriou, E. Kourti, C. Christodoulou, and V. Gkania, "Dynamic estimation of optimal dispatching locations for taxi services in mega-cities based on detailed GPS information," *IFAC (International Federation of Automatic Control) Conference*, vol. 49, no. 3, pp. 197–202, 2016.
- [10] M. Ramezani and M. Nourinejad, "Dynamic modeling and control of taxi services in large-scale urban networks: a macroscopic approach," *Transportation Research Procedia*, vol. 23, pp. 41–60, 2017.
- [11] M. Kabak, M. Erbaş, C. Çetinkaya, and E. Özceylan, "A GIS-based MCDM approach for the evaluation of bike-share stations," *Journal of Cleaner Production*, vol. 201, pp. 49–60, 2018.
- [12] J. Kim and H. S. Mahmassani, "Spatial and temporal characterization of travel patterns in a traffic network using vehicle trajectories," *Transportation Research Part C: Emerging Technologies*, vol. 59, pp. 375–390, 2015.
- [13] P. D'Urso, L. D. Giovanni, and R. Massari, "Robust fuzzy clustering of multivariate time trajectories," *International Journal of Approximate Reasoning*, vol. 99, pp. 12–38, 2018.
- [14] L. Zheng, D. Xia, X. Zhao et al., "Spatio-temporal travel pattern mining using massive taxi trajectory data," *Physica A: Statistical Mechanics and its Applications*, vol. 501, pp. 24–41, 2018.
- [15] J. J. Tang, W. Bi, F. Liu, and W. H. Zhang, "Exploring urban travel patterns using density-based clustering with multi-attributes from large-scaled vehicle trajectories," *Physica A*, vol. 561, pp. 1–18, 2020.
- [16] L. Ni, X. Wang, and X. Chen, "A spatial econometric model for travel flow analysis and real-world applications with massive mobile phone data," *Transportation Research Part C: Emerging Technologies*, vol. 86, pp. 510–526, 2018.
- [17] S. Zhang, X. Liu, J. Tang, S. Cheng, Y. Qi, and Y. Wang, "Spatio-temporal modeling of destination choice behavior through the Bayesian hierarchical approach," *Physica A: Statistical Mechanics and its Applications*, vol. 512, pp. 537–551, 2018.
- [18] M. Kamruzzaman, J. De Vos, G. Currie, B. Giles-Corti, and G. Turrell, "Spatial biases in residential mobility: implications for travel behaviour research," *Travel Behaviour and Society*, vol. 18, pp. 15–28, 2020.
- [19] J. Tang, J. Liang, S. Zhang, H. Huang, and F. Liu, "Inferring driving trajectories based on probabilistic model from large scale taxi GPS data," *Physica A: Statistical Mechanics and its Applications*, vol. 506, pp. 566–577, 2018.
- [20] J. J. Vázquez, J. Arjona, M. Linares, and J. Casanovas-Garcia, "A comparison of deep learning methods for urban traffic forecasting using floating car data," *Transportation Research Procedia*, vol. 47, pp. 195–202, 2020.
- [21] F. Zong, Y. Tian, Y. He, J. Tang, and J. Lv, "Trip destination prediction based on multi-day GPS data," *Physica A: Statistical Mechanics and its Applications*, vol. 515, pp. 258–269, 2019.
- [22] P. Krishnakumari, H. van Lint, T. Djukic, and O. Cats, "A data driven method for OD matrix estimation," *Transportation Research Part C: Emerging Technologies*, vol. 113, pp. 38–56, 2020.
- [23] Z. Chan, L. Collins, and N. Kasabov, "An efficient greedy K-means algorithm for global gene trajectory clustering," *Expert Systems with Applications*, vol. 30, no. 1, pp. 137–141, 2006.
- [24] R. Liu, H. Wang, and X. Yu, "Shared-nearest-neighbor-based clustering by fast search and find of density peaks," *Information Sciences*, vol. 450, pp. 200–226, 2018.
- [25] K. Kailing, H. P. Kriegel, A. Pryakhin, and M. Schubert, "Clustering multi-represented objects with noise," in *Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, Sydney, Australia, May 2004.
- [26] L. Gong, T. Yamamoto, and T. Morikawa, "Identification of activity stop locations in GPS trajectories by DBSCAN-TE method combined with support vector machines," *Transportation Research Procedia*, vol. 32, pp. 146–154, 2018.
- [27] D. Zhang, K. Lee, and I. Lee, "Hierarchical trajectory clustering for spatio-temporal periodic pattern mining," *Expert Systems with Applications*, vol. 92, pp. 1–11, 2018.
- [28] B. A. Sabarish, R. Karthi, and T. G. Kumar, "Graph similarity-based hierarchical clustering of trajectory data," *Procedia Computer Science*, vol. 171, pp. 32–41, 2020.
- [29] A. L. T. Palma, *A Clustering-Based Approach for Discovering Interesting Places in Trajectories*, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, 2008.
- [30] K. P. Agrawal, S. Garg, S. Sharma, and P. Patel, "Development and validation of OPTICS based spatio-temporal clustering technique," *Information Sciences*, vol. 369, pp. 388–401, 2016.
- [31] P. Das, D. K. Das, and S. Dey, "A modified Bee Colony Optimization (MBCO) and its hybridization with k-means for

- an application to data clustering," *Applied Soft Computing*, vol. 70, pp. 590–603, 2018.
- [32] T. Wu, Y. Zhou, Y. Xiao, D. Needell, and F. Nie, "Modified fuzzy clustering with segregated cluster centroids," *Neuro-computing*, vol. 361, pp. 10–18, 2019.
 - [33] G. Wang, C. Bu, and Y. Luo, "Modified FDP cluster algorithm and its application in protein conformation clustering analysis," *Digital Signal Processing*, vol. 92, pp. 97–108, 2019.
 - [34] D. Ienco and G. Bordogna, "Fuzzy extensions of the DBScan clustering algorithm," *Soft Computing*, vol. 22, no. 5, pp. 1719–1730, 2016.
 - [35] N. G. Seresht, R. Lourenzutti, and A. R. Fayek, "A fuzzy clustering algorithm for developing predictive models in construction applications," *Applied Soft Computing*, vol. 96, Article ID 106679, 2020.
 - [36] M. R. Mahmoudi, D. Baleanu, Z. Mansor, B. A. Tuan, and K.-H. Pho, "Fuzzy clustering method to compare the spread rate of Covid-19 in the high risks countries," *Chaos, Solitons & Fractals*, vol. 140, Article ID 110230, 2020.
 - [37] C. Wu and Y. Chen, "Adaptive entropy weighted picture fuzzy clustering algorithm with spatial information for image segmentation," *Applied Soft Computing*, vol. 86, Article ID 105888, 2020.
 - [38] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.
 - [39] S. Manochandar, M. Punniyamoorthy, and R. K. Jeyachitra, "Development of new seed with modified validity measures for k-means clustering," *Computers & Industrial Engineering*, vol. 141, Article ID 106290, 2020.
 - [40] F. Liu and Z. Zhang, "Adaptive density trajectory cluster based on time and space distance," *Physica A: Statistical Mechanics and its Applications*, vol. 484, pp. 41–56, 2017.
 - [41] R. M. Aliguliyev, R. M. Aliguliyev, and L. V. Sukhostat, "Weighted consensus clustering and its application to Big data," *Expert Systems with Applications*, vol. 150, Article ID 113294, 2020.
 - [42] H. P. Kriegl and M. Pfeifle, "Density-based clustering of uncertain data," in *Proceedings of the KDD Conference*, Chicago, IL, USA, August 2005.
 - [43] S. Barak and T. Mokfi, "Evaluation and selection of clustering methods using a hybrid group MCDM," *Expert Systems with Applications*, vol. 138, Article ID 112817, 2019.
 - [44] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 2, pp. 224–227, 1979.
 - [45] T. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Communications in Statistics - Simulation and Computation*, vol. 3, no. 1, pp. 1–27, 1974.
 - [46] R. A. Jarvis and E. A. Patrick, "Clustering using a similarity measure based on shared near neighbors," *IEEE Transactions on Computers*, vol. C-22, no. 11, pp. 1025–1034, 1973.
 - [47] D. Malchiodi, S. Bassis, and L. Valerio, "Discovering regression data quality through clustering methods," in *Proceedings of the New Directions in Neural Networks-18th Italian Workshop on Neural Networks*, pp. 76–85, IOS Press, Amsterdam, Netherlands, July 2008.