CAAI Transactions on Intelligence Technology

The Institution of Engineering and Technology WILEY

Survey on vehicle map matching techniques

Zhenfeng Huang[1] | **Shaojie Qiao**[1] | **Nan Han**[2] | **Chang-an Yuan**[3,4] |
Xuejiang Song[5] | **Yueqiang Xiao**[2]

[1]School of Software Engineering, Chengdu University of Information Technology, Chengdu, China

[2]School of Management, Chengdu University of Information Technology, Chengdu, China

[3]School of Computer and Information Engineering, Nanning Normal University, Nanning, China

[4]Guangxi College of Education, Nanning, China

[5]Chengdu Tanmer Technology Co., Ltd, Chengdu, China

Correspondence

Shaojie Qiao, School of Software Engineering, Chengdu University of Information Technology, Chengdu, China.
Nan Han, School of Management, Chengdu University of Information Technology, Chengdu, China.
Email: sjqiao@cuit.edu.cn, hannan@cuit.edu.cn

Abstract

With the development of location-based services and Big data technology, vehicle map matching techniques are growing rapidly, which is the fundamental techniques in the study of exploring global positioning system (GPS) data. The pre-processed GPS data can provide the guarantee of high-quality data for the research of mining passenger's points of interest and urban computing services. The existing surveys mainly focus on map-matching algorithms, but there are few descriptions on the key phases of the acquisition of sampling data, floating car and road data preprocessing in vehicle map matching systems. To address these limitations, the contribution of this survey on map matching techniques lies in the following aspects: (i) the background knowledge, function and system framework of vehicle map matching techniques; (ii) description of floating car data and road network structure to understand the detailed phase of map matching; (iii) data preprocessing rules, specific methodologies, and significance of floating car and road data; (iv) map matching algorithms are classified by the sampling frequency and data information. The authors give the introduction of open-source GPS sampling data sets, and the evaluation measurements of map-matching approaches; (v) the suggestions on data preprocessing and map matching algorithms in the future work.

1 | INTRODUCTION

In this era of Big data, a huge volume of data is being produced every day, which has been utilized to conduct the research studies. After extracting and filtering the data, researchers have used specific rules and designed mathematical models to analyze and discover knowledge from the data [1] in order to obtain useful information for people's daily activities. With the rapid development of location-based services, trajectory data [2,3] are growing exponentially, which represent the continuous movement of objects, including information such as latitude and longitude. These continuously growing trajectory data assist in the intelligent transportation in several application scenarios. As shown in Figure 1(a), trajectory data can serve the upper application layer in the research of intelligent transportation, for example taxi passenger points of interest mining [4], updating urban road network [5], vehicle abnormal behaviour detection [6] and ride-sharing taxi scheduling [7]. Inevitably, the first step is to correct the deviation of trajectory data with map-matching techniques. Then, the corrected data is analyzed and applied to the application layer. In addition, the data generated by the upper application can also be used in the data layer. In practice, there is a certain deviation between the actual driving routes and global positioning system (GPS) sampling data because of the satellite positioning error [8]. In addition, "Since the vehicle navigation system was proposed [9–11], the problems of inaccurate positioning and system abnormality have always existed. In addition, several scholars reported that the navigation system initially needs to perform GPS point correction on the electronic map so that the navigation function can be used effectively [12]." In the research of map matching, a Kalman filter is proposed for vehicle navigation [13], which mainly uses the covariance matrix to determine the specific location of the vehicle when the signal is blocked, thus completes the map-matching process of the vehicle position. Hereafter, based on the Kalman filter, a multihypothesis technique for map matching is proposed based on the principle that the likelihood function is used to infer the target position of the vehicles [14]. Figure 1(b) describes the workflow of the vehicle intelligent navigation system. Intuitively, after the map-matching step, the navigation function can work properly, such as route guidance, velocity control, route visualization and position location. As shown in Figure 1, map-matching technique is the core of navigation technology and plays an essential role in the field of intelligent transportation and intelligent navigation system [15,16].

In recent years, with the rapid development of wireless network technology, map-matching techniques have wide application scenarios, such as indoor navigation [17], unmanned aerial vehicle flight [18], robot training [19] and so on. Therefore, it is necessary to use map-matching techniques to rectify the deviation of positions w.r.t. trajectory data. As for the use of trajectory data, we focus on the floating car data (FCD) collected by moving vehicles in the map-matching environment due to its popularity in transportation systems.

(a)

Upper application layer

| Taxi passenger POI mining | Vehicle abnormal behavior detection |
| Updating urban road network | Ride-sharing taxi scheduling |

Data analysis

Map matching

Data layer

Trajectory data   Road network data

Research in intelligent transport system

(b)

Navigation function

| Route guidance | Velocity control |
| Route visualization | Position location |

Map matching

Sampling data

Trajectory data   Road network data

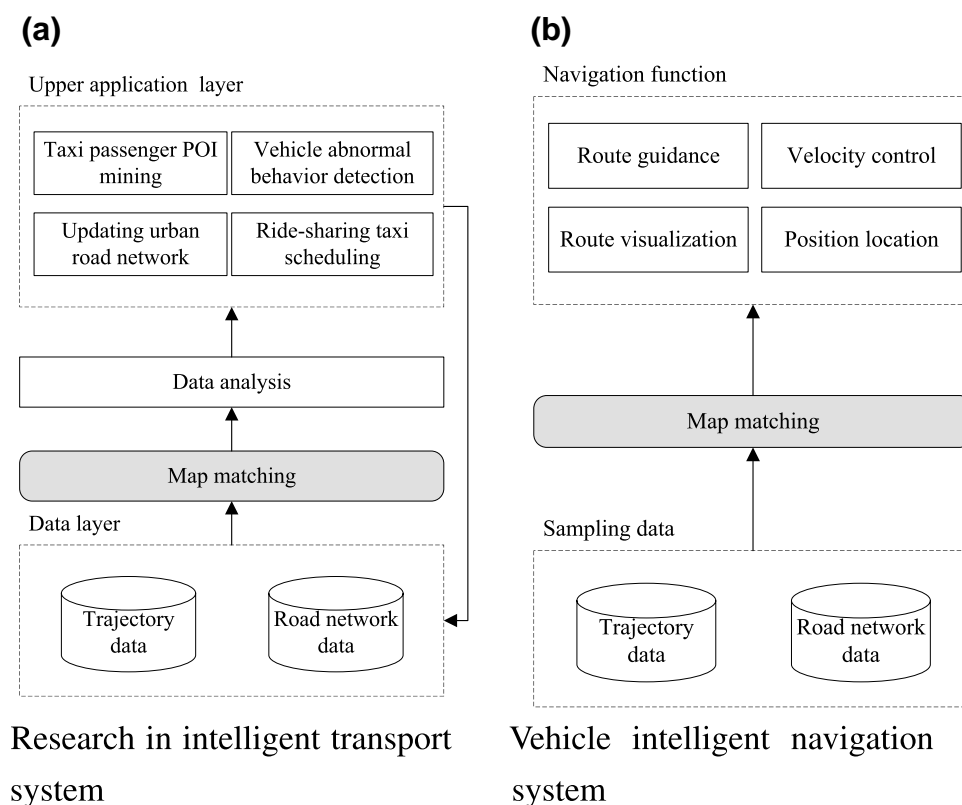Vehicle intelligent navigation system

FIGURE 1 Application scenario of map-matching techniques

As shown in Figure 2, map-matching technique is a complete process including FCD preprocessing, road data preprocessing and selection algorithms [20]. The phase of data preprocessing aims to process the original FCD and road data by using many different kinds of methods to obtain the ideal experimental results. The phase of selection algorithms focuses on selecting an appropriate one among many algorithms according to the actual situation. In the phase of map matching, the task is to design appropriate map-matching algorithms on the processed experimental data. In the practical applications, in the absence of such a process, it will have a significant impact on the experimental results. The phase of visualized results output is an additional step after map matching to visualize the experimental results. By using geographic information system (GIS) tools [21], we can find whether the trajectory data are correctly mapped to the actual path in the electronic map.

The research method of software engineering is very popular, which needs to establish the review goal and research method, and then draws a conclusion [22–24]. In [25], researchers believed that a systematic framework can effectively assign effective plans and strategies to assist practitioners. Ghazi et al. [26] reported that the demand for empirical research on software engineering is growing, and by following the research methods of software engineering to determine the problems researchers face in the investigation design and mitigation strategies. However, in order to comprehensively review the map-matching techniques, we review map-matching techniques from a new perspective, including the data preprocessing, selection algorithms, the open source data sharing and evaluation measurements. Starting off the point of view of technology, we aim to summarize the original FCD and road data preprocessing rules, and classify map-matching algorithms from different aspects. Overall, our contributions are as follows in this survey: (i) We give a detailed review of the data preprocessing process, which is rarely mentioned in the map-matching problems, and operation methods and significance of the rules are also presented in the Tables 1, 2 and 3; (ii) We categorize the map-matching algorithms according to the GPS sampling frequency and the data information; (iii) Open source trajectory data sets and common evaluation measurements are provided. We do not only describe the data sets, but also provide their downloading websites. In addition, we have made the necessary explanation of the evaluation measurements and (iv) Based on the current trend of map-matching techniques, we provide the future development direction.

The rest of the study is organized as follows: In Section 2, we first provide the necessary definition of map matching and analyze the existing research work. Then, a methodology is given in Sections 3 and 4. Finally, we propose the future direction of map matching and give conclusion in Section 5.
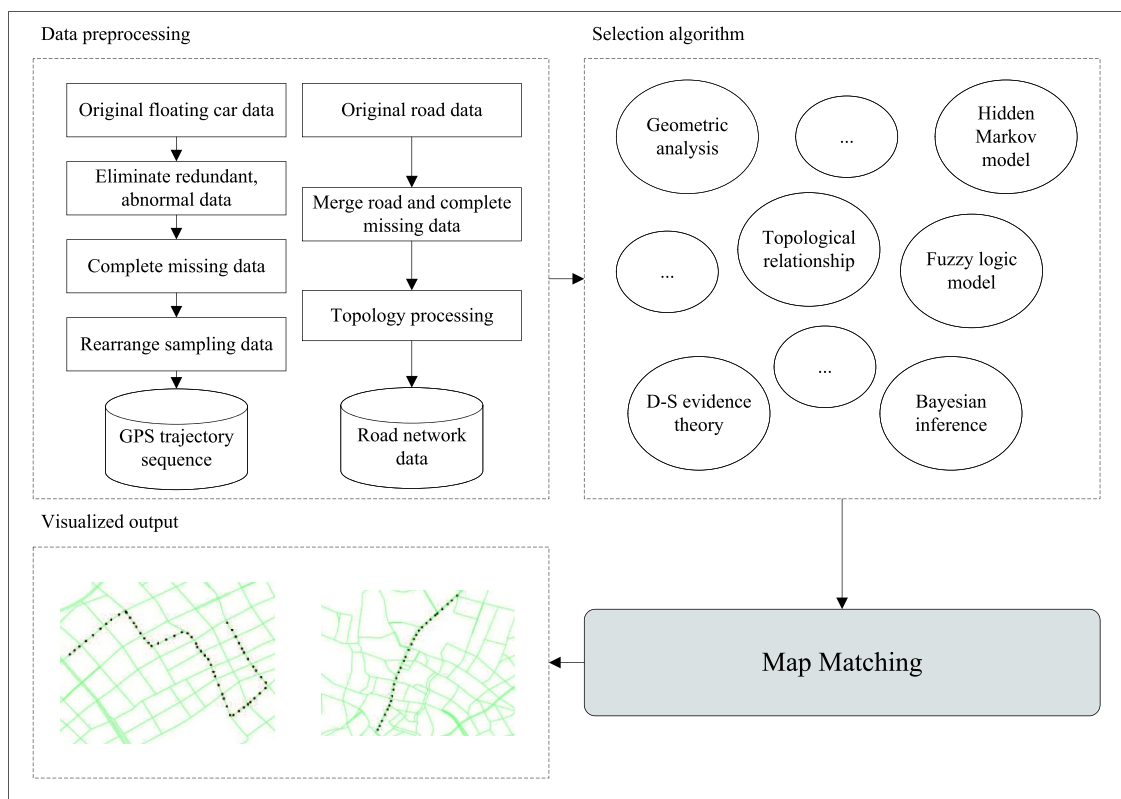

FIGURE 2 Diagram of map-matching techniques

TABLE 1 Preprocessing rules of noise filtering

Rules	Operation methods	Significance	References
Filtering redundance	Set a sliding window, and delete the latter same piece of data inside the window.	Delete redundant data to avoid duplicate points for subsequent map matching.	[36–40]
Removal of abnormality	Set the scope of trajectory areas and remove the trajectory points beyond the trajectory area. Setting speed thresholds to filter abnormal data is also popular.	Remove meaningless points to avoid increasing the matching error.	[36–42]
Removing incomplete data	Delete missing records, or delete attributes corresponding to missing values. Data with latitude and longitude with value zero also should be removed.	Make the data more reasonable.	[43–46]

TABLE 2 Preprocessing rules of machining data

Rules	Operation methods	Significance	References
Coordinate transformation	Convert the WGS184 coordinate system into other coordinate system, such as Beijing 50. The main task is the conversion of parameters between different coordinate systems	Make the experimental data in the same projection coordinate system	[47–50]
Filling missing values	The average method aims to find the mean value. The median method focuses on using median value. Linear interpolation is mainly to use a linear formula for the previous point to calculate the next point. Forward and backward interpolation is directly filled with adjacent data. Temporal interpolation focuses on time series interpolation. Multiple filling method needs to predict different values	Make full use of the missing value data on the map matching in order to improve the map-matching accuracy	[43, 51–54]
Reconstruction trajectories	Combining with the inference model and the existing trajectory information to restore the target trajectory route	The reconstructed trajectory is closer to the actual driving route and reduces the difficulty of map matching	[41, 55–58]
Trajectory segmentation	The long trajectories are segmented according to certain rules of trajectory distance, time period, geometric topology, trajectory semantics etc.	Segmental trajectory facilitates parallel map matching	[28, 35, 59–61]
Geometric information indexing	Set index mark for position information. After the map is meshed, it also can quickly locate each grid	Accelerating the search of trajectory points in map matching	[35, 62–65]
Data arrangement	Arrange the data according to the time stamps	Restore messy data to the original trajectory order	[48, 66–68]

TABLE 3 Preprocessing rules of road data

Rules	Operation methods	Significance	References
Road extraction	Road extraction should be carried out according to road data distribution and experimental environment.	Extract the required road as the experimental data.	[69–73]
Road correction	Set a threshold to determine whether the two segments need to merge.	Ensure the correctness of road data.	[69, 74-77]
Grid partition	Divide the map into grids of the same size. You can customize the mesh size according to the requirements.	The purpose of grid partition is to speed up the search of candidate segments.	[78–82]
Topological construction	The structure of the 'node-segments' is popular. You can use GIS tools to build topology on roads.	Topology can better express the connection relationship between roads.	[83–87]

2 | PRELIMINARY

2.1 | Problem definition

Definition 1 (FCD) In Figure 3, given a set of FCD $T = (P_1, P_2, P_3, \ldots, P_n)$, where each point P_i is a GPS trajectory point generated by the order of timestamps, and each P_i includes the information such as ID number, latitude and longitude, time, vehicle angle, vehicle speed etc. The three basic elements of FCD include longitude, latitude and time information. The longitude and latitude represent the position of the

Floating car data

	ID	Longitude	Latitude	Time	Speed	Angle	Status
P_1:	116	121.594800	31.200000	2007-02-20 00:00:10	62	135	1
P_2:	116	121.583500	31.198000	2007-02-20 00:01:11	80	135	1
...	116	...	...	...	...	...	1
P_n:	116	121.547800	31.190500	2007-02-20 00:12:56	52	45	0

GPS trajectory
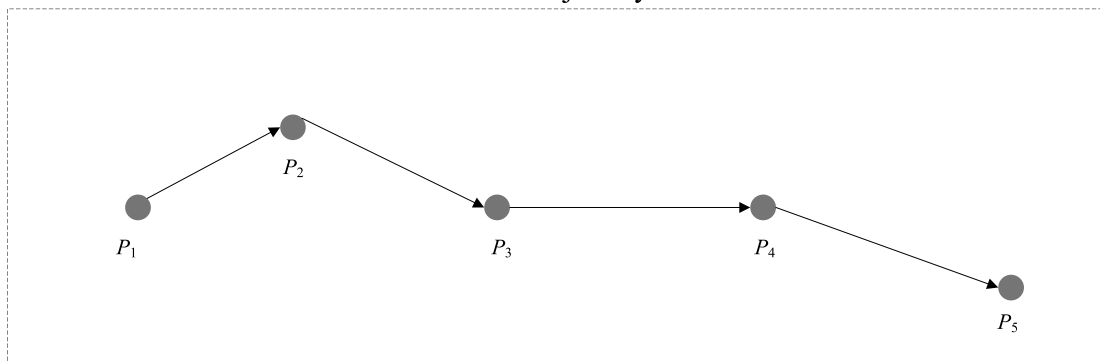
FIGURE 3 Description of FCD

FIGURE 4 Road network structure

FIGURE 5 Candidate segment
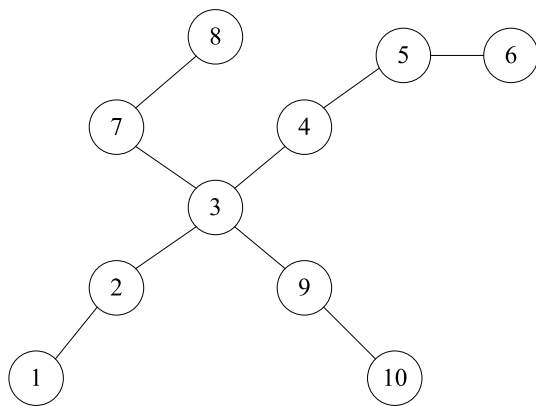
vehicle, and the GPS trajectory of the vehicle can be determined according to the time information. The more fields FCD contains, the more conditions the matching algorithms can take into consideration, and the higher is the matching accuracy.

Definition 2 (Road network structure) Because of the complexity of urban road network structure, the topology model can be used to clearly express the relationship of roads [27]. The original road data must be constructed in to a topological graph, please refer to Section 3.2. Assuming that the point is a road junction and the line represents a road segment, then all road
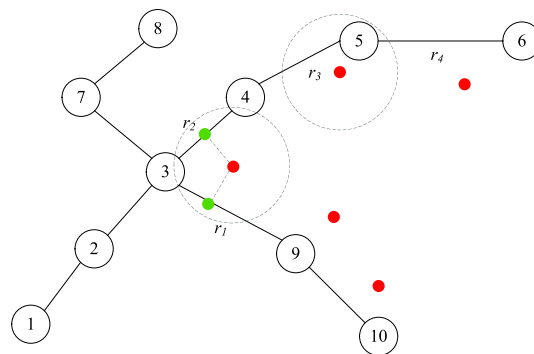
network structures can be represented as shown in Figure 4 and Figure 5. For example, there are two interlaced roads. One road is called 'Beijing Road', which contains five road segments and six road junctions numbered from 1 to 6. Another road is called 'Nanjing Road', which consists of four road segments and five road junctions with intersection numbers 3, 7, 8, 9 and 10. Straightforwardly, the intersection of 'Beijing Road' and 'Nanjing Road' is the third road junction. Given a urban road network data, it can be expressed by an undirected graph network G = (V, E), where V is a set of vertices representing road segments intersection, E is a set of edges representing the road segments. In the phase of road data preprocessing, the road segments

can be stored by the methods proposed in [28,29], while the road segments intersection is stored by the method introduced in [30].

Definition 3 (Candidate segment) In map matching, we need to find several segments as candidate objects for trajectory points, which are likely to be the actual path where the trajectory points are located. Typically, we set the search radius centred on the current trajectory point and search candidate segment within circular regions. The *red dot* is the trajectory point, and the circle is the search area. Trajectory segments searched by trajectory points include r_1, r_2, r_3 and r_4. Correspondingly, the *green dot* is the matching point of the candidate segments, also known as the candidate matching point.

2.2 | Related work

Intuitively, when the vehicle is driving on the road, a complete trajectory should be correctly displayed on the road. However, due to the errors caused by satellite positioning systems, there are deviations between vehicle trajectory and roads. The map matching problem has been extensively studied in recent 2 decades, but there are few surveys of map-matching techniques, and most of existing work focused on designing map-matching algorithms.

From a methodological perspective, Quddus et al. [31] gave a review of map-matching approaches, including geometric analysis, topological analysis and probabilistic map-matching and advanced map-matching algorithms. Geometric analysis utilizes the distance relationship between trajectory points and roads for map matching. Topological analysis focuses on the connectivity of trajectories and adjacency road. The probability-based map-matching algorithm uses different mathematical models to find a trajectory with the maximum probability approximating to the real path. The advanced map-matching algorithm can reduce positioning errors from multiple aspects, including Kalman filter, fuzzy logic, multiple hypothesis technique and so on. Based on the classification method proposed in [31], Hashemi et al. [32] classified the real-time map-matching algorithms. A variety of different matching algorithms are compared and analyzed. In addition, the disadvantages of weight-based and advanced map-matching algorithms are also compared and analyzed. Song [33] introduced a more comprehensive classification method of map-matching algorithms, which is categorized according to information, application environment and sampling frequencies. Different from previous classification methods, Chao et al. [34] proposed an innovative classification method based on matching models and real scenarios. Similarity measurements, state-transition, candidate-evolving and scoring model are proposed in the study. The similarity model contains two types, that is, distance based and pattern based, corresponding to the distance of the points or trajectory segments projected to the nearest road. Based on the weighted topological graph of all possible vehicle driving routes, the main task of the state-transition model is to find an optimal path, such as hidden Markov model, conditional random field and weighted graph technique. The candidate-evolving model needs to maintain a set of candidates in the phase of map matching, and evolves from the first point by constantly adding new candidates while pruning irrelevant ones. The particle filter and multiple hypothesis techniques are the most classical methods. The scoring model first needs to partition road into grids, scores the grids according to the timestamp and returns the maximum score of the grid results. Taking into consideration the noise characteristics of trajectory data, Gao et al. [35] introduced the trajectory data preprocessing methods including data cleaning, trajectory compression, trajectory segment and map matching. Noise filtering and stay point detection methods can effectively remove outliers in the phase of data cleaning. However, in order to reduce storage capacity and improve the efficiency of calculation, both distance-based and offline compression methods can work well in trajectory compression. Trajectory segmentation is beneficial for reducing the computational complexity, which is commonly used in three strategies based on time threshold, geometric topology and trajectory semantics. In the phase of map matching, the processed trajectory data and electronic map are combined to transform the sampling sequences of GPS coordinates into the road network coordinates. Real-time online matching and offline matching for historical static data are two most popular approaches.

Although map-matching problem has been studied for many years, many survey studies have deficiencies in some aspect. For example, many researchers regard map matching as map-matching algorithms problem, ignoring the data preprocessing process. Data preprocessing is the initial phase of map matching and should be explored in detail. In addition, the classification of existing map-matching algorithms is not state-of-the-art, so it is necessary to propose a new classification method. These studies in [31–33] just focus on different classification of map-matching algorithms, but it is not a comprehensive study of map matching. In addition, open source GPS sampling data sets and the evaluation measurements are rarely mentioned in the map-matching surveys. Although open source GPS sampling data sets and the evaluation measurements are not the main research contents of the study, we believe that the map-matching surveys should provide more relevant information to facilitate reader's understanding. In [34,35], it should provide more detailed information. Different from the existing work, we first make a comprehensive summary of data preprocessing, including FCD preprocessing and road data preprocessing. Then, we propose a new classification method for map matching. Finally, open source GPS sampling data sets and the evaluation measurements. In conclusion, we summarize the map-matching problem as comprehensively as possible.

In this section, we first give the necessary definition of map-matching techniques. Then, after a comprehensive review of the existing work, we propose a new classification method for map-matching techniques.

3 | SURVEY OF DATA PREPROCESSING

3.1 | FCD preprocessing
3.1.1 | Noise filtering

The results obtained by using the original FCD with noise directly in experiments are often not ideal. In the process of data acquisition, due to the frequency of acquisition equipment or acquisition rules, it is inevitable to produce records that are regarded as noise data, such as redundant, abnormal and incomplete records. Here, we will give the following introduction to the noise data filtering approaches as shown in Table 1.

Filtering redundance. When the speed is slow or the vehicle stops, the trajectory points may overlap and the redundant data will generate. Yan et al. [36] combined the speed and distance between adjacent points to determine whether there are redundant points. If the adjacent point distance does not exceed the threshold and the speed is less than 3 m/s, the current trajectory point is ignored. Huo et al. [37] thought that the probability of GPS producing the same record in a short time is very small, so the same record is deleted within a fixed time interval. Hui et al. [38] assumed that no more than 50 m of vehicle position change within 30 min was a parking behaviour and deleted redundant sampling points with the same timestamp. GPS records may have data with the same time and different speeds, so only a set of data needs to be retained in these studies [39,40].

Removal of abnormality. When an extreme situation occurs, the data will be abnormally generated. For example, if the visible satellite is less than 4 or the vehicle is more than 200 km/h, they are considered as abnormal data and be removed in [41]. In the study [36], FCD beyond the limitation of longitude and latitude is regarded as abnormal data and should be removed, which can only appear within the fixed area of the electronic map, because there are more stringent speed limitations for urban roads, generally 40 km/h. It is only necessary to determine the two coordinates of lower left corner and top right-hand corner which are viewed as the latitude and longitude limitations of the electronic map. Liu et al. [40] believed that the travel time conformed to a log-normal distribution and used the Laida criterion to clean outliers. Wen et al. [39] deleted data with a speed of 0 because they could not reflect the driving behaviour. Wang et al. [42] filtered data with negative speed and replaced it with zero for subsequent adaptive tracking of vehicles.

Removing incomplete data. Ye et al. [43] considered that any records containing missing values should be removed, and only use the complete records. When the attribute variable of one column of the data has a missing value, the entire column attribute is deleted. Although attribute deletion can impact the integrity of the data sets, this method is very effective when the amount of data deleted in the whole data sets is very small. From the viewpoint of coordinates, Dong et al. [44] removed the data with latitude and longitude with value zero. The missing records are deleted in these studies [45,46] because they lack valid information and are ineffective.

3.1.2 | Machining data

Different machining methods can be selected for raw data according to the actual situation so that the map-matching algorithms can obtain a positive result. Note that not all machining methods can be applied in the real data. It is a good way to choose the machining method according to the structure of the data. As shown in Table 2, we summarize some rules of machining data.

Coordinate transformation: In order to reduce the conversion error of latitude and longitude data received by GPS navigation systems in the phase of transformation in electronic map, based on the seven-parameter transformation model, Tan et al. [47] analyzed the effects of the distribution of different public point positions on the transformation accuracy to establish the relationship between the distribution density of data and the transformation accuracy. The key of coordinate transformation is to solve the transformation parameters and transform them into the coordinate parameters of the corresponding region. Zhang et al. [48] believed that there are positioning errors in different coordinate projection systems. They transformed GPS WG184 coordinate system into Beijing 50 or Xi'an 80 coordinate system using the transformation of spatial coordinate systems. Li et al. [49] first transformed WGS-84 spatial coordinates into BJ-54 spatial coordinates, and finally determined the plane coordinates by Gaussian projection. Zhang et al. [50] believed that there would be errors in converting GPS measurement results to BJ-54 or Xi'an 80 coordinates, and proposed to use Gauss–Kruger projection to convert to plane coordinates.

Filling missing values: Missing value, as a factor to improve the quality of data mining, cannot be ignored. In literature [51], the five filling methods are summarized, by mainly using interpolation and filling function. Specific methods are given as follows: (a) average method: replace missing values with the average value of the entire sequences; (b) median method: use the median of a certain span range or all valid values to replace the missing values; (c) linear interpolation: the data of the previous position and the latter position of the missing value are interpolated by using a linear formula; (d) forward and backward interpolation: fill in the missing value with the previous or the latter value and (e) temporal interpolation: apply the closest value in time distance to the linear model in order to fill the missing values. Ye et al. [43] believed that the method of multiple filling has more advantages than single values filling, which aims to construct multiple estimates for each missing value and form several complete data sets. The method can simulate the distribution of missing data as close as possible to the real situation and maintain the original relationship between variables. Cai et al. [52] formulated the trajectory missing distance and time rules, and interpolated the missing trajectory points according to the missing number equal distance. Yin et al. [53] proposed a model using long short-term memory networks to solve the problem of poor effectiveness of traditional completion methods. By comparing various interpolation methods, Wang et al. [54] found that regularized algorithm has better interpolation effect.

Reconstruction trajectories: When the acquisition time interval is longer or the signal shielding occurs, the phenomenon of discontinuity will occur in the trajectory points. In order to improve the unity and standard of the trajectory data, and at the same time to obtain the complete motion trajectory of the vehicle by using the limited trajectory data, Sheng et al. [55] designed an improved Hermite interpolation algorithm to perform temporal interpolation on the trajectory points after modifying the errors. The key task of this method is to use polynomial function to restore the trajectories and add restrictions on the vehicle speed and driving distance. Zeng et al. [41] compensated the data using dead reckoning method, which uses the latitude, longitude, velocity, angle and positioning time interval of the previous point to calculate the position of the next point. Zhao et al. [56] considered that the error between measurement points has an effect on the reconstructed trajectory, and combined the 3σ criterion with the inverse distance weighted interpolation method to generate a new trajectory. Shi et al. [57] used Lagrangian fifth and third polynomial interpolation to estimate the trajectory points, and then used Kalman filter algorithm to remove the noise of the trajectory. Wang et al. [58] first calculated the statistical information of driving mode with probability distribution, then used maximum likelihood estimation to reconstruct the trajectory.

Trajectory segmentation: By dividing the trajectory into several segments, the data can be displayed in detail and clearly, which are especially beneficial for the heuristic matching algorithms. Zhang et al. [28] made the rule that the trajectory does not intersect and the distance of the trajectory is greater than the threshold, and partition the trajectory into multiple segments without loops. According to the speed variables, Yang et al. [59] divided the long-time sequence trajectory into three states: high speed, medium speed and low speed. The partitioned trajectory segments need to be used to find homomorphic trajectory and merge into subtrajectories. In addition, partitioning methods based on time threshold, geometric topology and trajectory semantic strategies are also mentioned [35]. Zhang et al. [60] used classification and regression tree to segment the original trajectory while calculating the synchronized Euclidean distance (SED) between adjacent trajectory points. Finally, the subtrajectories are merged and simplified by SED. Zhang et al. [61] calculated the change of velocity and direction of adjacent points. When the threshold is exceeded, the trajectory is segmented.

Geometric information indexing: Finding a trajectory directly on an electronic map is difficult. Geographic location information is a tuple containing longitude and latitude that can be converted into other expressions for localization. Kang et al. [62] used the Geohash technique to encode the longitude and latitude of the trajectory points in order to find the candidate segments. This method can only be used to search the nearby nine mesh information so that greatly reduces the time complexity. Gao et al. [35] introduced the index structure such as R-tree, B-tree and K-D tree. In addition, some researchers proposed to use the method of griding to index the trajectory data [63,64]. Hong et al. [65]

used MapInfo software to establish a spatial index search based on a single-layer grid, using two staggered partition mesh division methods.

Data arrangement: Sorting data is necessary because processed data can sometimes appear complex. There is only one popular way to sort the trajectory data, that is, to arrange the data according to the time stamps. The study [48,66,67] sorted the data in the ascending order according to the sampling time, where the trajectory of each vehicle is from the departure to the destination. Nevertheless, Zhong et al. [68] sort the trajectory points according to the principle of nearest neighbour distance to form the trajectory with sequence.

3.2 | Road data preprocessing

Road data are necessary for map matching because the trajectory points are ultimately to be displayed on an electronic map containing road data. Most electronic maps consist of road data, such as 'Baidu Map', 'Google Map' and so on. Inevitably, road data also appear missing and wrong, due to slow road renewal and staff mapping errors. For FCD, road data also need to be preprocessed. As shown in Table 3, we summarize the methods of road data processing.

Road extraction: The goal is to extract the required path. Luo et al. [69] thought that the path planning algorithm needs to work in a network of road connectivity, and combined high-rise and low-rise networks to ensure connectivity including national highway, provincial road, county road, highway and so on. Zhang et al. [70] used the regular expressions to quickly match the node information of road files to obtain the road networks. If the road contains multiple segments, each segment is treated as an object. Yu et al. [71] aimed to explore common road network models and constructed five simplified road network models, such as rectangle and tree branch. Huang et al. [72] extracted road data such as highway and ramp, and carried out operations such as data duplication, data integration and data segmentation. Zhang et al. [73] classified OpenStreetMap data through tags, and then used regular expressions to match node information to obtain multiple road objects.

Road correction: Before constructing the topological graph of road data, we need to check whether the road data are correct. The work done in [69] has focused on repairing road separation. After setting the distance threshold, when the distance between the two end points of the separated segments is less than the distance threshold, the road is merged. Considering the disadvantages of map information in electronic maps, Sun et al. [74] proposed a buffer analysis method to repair the road data by specifying the distance threshold. Instead of changing the raw road data, they added supplementary information to the property tables to generate new layers. In order to meet the needs of traffic management business, Wang et al. [75] simplified the original road data. They perform multilane fragment removal, off-ramp removal on highways and multilane merging. Zeng et al. [76] deleted the pseudo nodes and separate arcs of the road data in order to obtain a connected road network. Deng et al. [77] believed that

the traditional method of extracting road intersections was not fine enough, and proposed a trajectory point clustering method to extract road intersections. This method uses local spatial autocorrelation analysis to detect trajectory steering angle hotspots, and uses adaptive spatial point clustering to cluster hotspots and identify the location of intersections.

Grid partition. Grid division focuses on equivalent block segmentation of maps. Fan et al. [78] partitioned the longitude and latitude of the map equally with 0.002 degree as the unit. Wang et al. [79] used the tree structure to divide the map step by step until it can no longer partition the grids. Li et al. [80] proposed to quickly and accurately locate the candidate road segments. They divided the road data into grid areas of the same size to build a grid index, using the ID number of grids to quickly identify all segments of the grids. Zhao et al. [81] proposed an adaptive basic grid division algorithm, which can effectively solve the problem of road search errors in equidistant grid division. Liu et al. [82] proposed to use multi-process technology to cut the original raster image to improve the speed of map division.

Topological construction. Topological construction aims to establish the expression relationship between roads. In [83, 84], they interrupted the road at the intersection point and divided the road into several small segments through the SuperMap GIS platform. In [85], road data can be partitioned by segments intersection into 'node-segments', and the topology is constructed by using the breadth-first traversal method. The work was done by Chen et al. [86], which proposed the dual structure of data storage layer as well as dynamic topological layer, and they mentioned that the topological relation of object space is dynamically changing. For expressing the relationship between segments and nodes, Li et al. [87] first recorded segments and nodes separately with tables, and then merged nodes with the same ID to construct road topology.

In this section, we first classify FCD preprocessing rules and introduce each processing rule in detail. Then, we summarize the rules of road data preprocessing.

4 | SURVEY OF SELECTION ALGORITHMS

As the map-matching techniques grow rapidly, algorithms based on geometric principle, topological structure, probability and machine learning are applied to map matching, which is also the key technique of map-matching techniques [88–91]. In this section, we classify map-matching algorithms with sampling frequency and data information. In addition, the open source GPS sampling data sets and evaluation measurements are given below.

4.1 | Classification of map-matching algorithms based on sampling frequency

In general, vehicles with GPS positioning devices record GPS information at a certain frequency. By considering energy
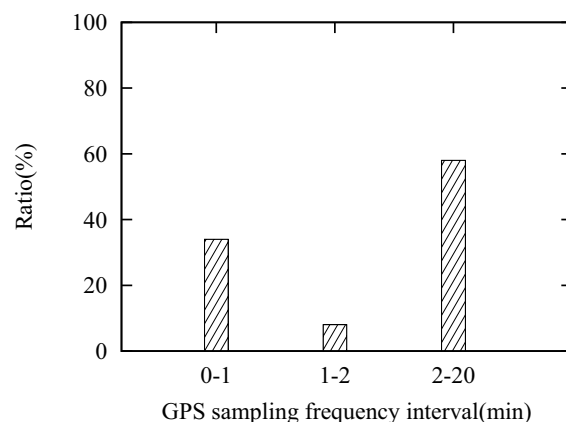

FIGURE 6 Statistics of sampling frequency of GPS data

TABLE 4 Classification of sampling frequency

Type of sampling	Sampling frequency
High-frequency sampling	1–10s
Low-frequency sampling	More than 30s
Ultra-low-frequency sampling	More than 2 min

consumption and communication cost, GPS information is continuously recorded at a lower frequency. As shown in Figure 6, Yuan et al. [92] counted the data generated by more than 10,000 taxis in Beijing for a week. By statistics, the data of sampling frequency within 1 min occupy 34% of the whole data, and it is regarded as high-frequency sampling data. Moreover, sampling frequency more than 1 min is considered to be low-frequency sampling data with a proportion of 66%. As described in Table 4, Gao et al. [93] viewed the sampling data between 1 s and 10 s as high-frequency sampling data, more than 30 s as low-frequency sampling data, and more than 2 min as ultra-low-frequency sampling data.

4.1.1 | High-frequency sampling data-based algorithm

High-frequency sampling data has the characteristics of dense points, small change of heading angle and smooth curve, which can be fully utilized to improve the accuracy of map matching. Based on the topological relationship and connectivity of the road network, Zeng et al. [94] played the advantage of dense trajectory points and used the shortest path search algorithm to obtain candidate road segments because taxi drivers preferred the shortest route. The more dense the trajectory points, the less candidate sections are searched by the trajectory points, and the higher the map-matching accuracy is. Chen et al. [95] used the relationship between the heading angle of the adjacent trajectory points and the candidate segments to find the road junction. If the angle is greater than 45 degrees, it is regarded as the road junction. Because the variation of heading angle of

TABLE 5 Classification of map-matching algorithms by sampling frequency

Classification	High-frequency sampling data-based algorithm	Low-frequency sampling data-based algorithm
Features of data sets	Dense trajectory points, small change of heading angle and smooth curve.	Trajectory points sparsity and large curve fluctuation.
Advantage	Dense trajectory points can be used to screen candidate segments and reduce computation. The advantage of small change in heading angle can be used to find road junctions. Curve smoothing preserves the integrity of the trajectory and reduces the difficulty of map matching. Because the trajectory smoothing preserves the integrity of the trajectory, map matching has a lower level of difficulty.	Map matching from the perspective of probability statistics will be better due to sparse trajectories.
Citations	[94–97]	[55, 92, 98, 99]

TABLE 6 Classification of the map-matching algorithms by data information

Algorithm	Citations	Principle	Scope
Geometric theory	[88]	Comparing the trajectory curve and the slope of the candidate segment	Incremental
	[100]	Construct reasonable search ranges to obtain candidate segment and then weight the distance and angle	Incremental
	[107]	Calculating the Frechet distance with discrete distance	Incremental
Topology	[113]	The trajectory is divided into segments without loops, and matching results are obtained by combining the road structure and custom rules	Incremental
	[87]	Build a topology on node, roads and intersections to achieve map matching using four different states.	Incremental
	[89]	The trajectory points are matched to subpaths by linear model	Incremental
Probability statistics	[108]	Set inference rules and weight them	Global
	[90]	Construct a search tree for the later selected sections and then evaluate the results using spatial and temporal information	Global
	[104]	The probability formula is used to search the candidate segment and the results are obtained by the correlation coefficient method	Global
	[105]	Using the Bayesian model to obtain the highest score path	Global
Advanced model	[109]	Using position and direction as reasoning evidence and establishing recognition function	Global
	[91]	Input information such as angle and position labels into the model and adjust the number of nodes in the hidden layer	Global
	[110]	Build the historical trajectory experience bases and use the reward function to find the optimal path	Global
	[111]	Using the weighted combination of spatial similarity, shortest path distance and direction similarity as the fitness function to output the optimal results	Global

high-frequency sampling data is small, the intersection can be found accurately in the experiment, but the performance effect is poor in the low-frequency sampling data. Liu et al. [96] proposed two different matching algorithms for the map with road segments. When considering the road complexity, the trajectory points are input into the algorithm model to get the results and feedback back to the model. Otherwise, the prediction results are obtained directly by fitting the formula. Both methods effectively solve the map matching of Y zigzag road. Since essentially considering the smoothness of the trajectory curve, high-frequency sampling data is required. Zhang et al. [97] concentrated on using weighted formulas for angle, distance and slope. If the ultra-low sampling frequency is used, there will be a 'zigzag' phenomenon.

4.1.2 | Low-frequency sampling data-based algorithm

Low-frequency sampling data has the disadvantages of point sparsity and large curve fluctuation, so map-matching algorithms generally focus on incremental, global or other aspects. Sheng et al. [55] found that the accuracy of the topology searching method decreases with the sampling time interval. They improved the Hermite interpolation method and used the single-zero and multi-zero-segment linear interpolation to obtain the matching results from the aspect of restoring the vehicle route. The concept of reliable points for screening candidate segments is proposed in [98]. By comparing the ratio of the top two candidate road segments of the transmitting probability row, the

remaining candidate road segments are removed when the ratio is greater than an empirical value, thus reducing the calculation of the transition probability. In addition, A^* algorithm and the sliding window are used for real-time matching. Research was done by Lou et al.[99], which proposed spatiotemporal (ST) matching algorithm that considers the spatio and temporal factors. The spatial analysis function takes into account the distance from the trajectory point to the road segments and the adjacent candidate matching points, while time analysis function combines the factors of road segments length and constraint speed. The product of the two functions is taken as the score of the candidate segments, and the segment with the highest score is taken as the matching result. Based on the work in [99], Yuan et al. [92] adopted the interactive voting approach. The candidate matching points need to vote with each other to eliminate the phenomenon of matching errors in adjacent segments. When compared with the ST-matching algorithm, the proposed algorithm is more robust. As shown in Table 5, we summarize the different characteristics of high-frequency and low-frequency sampling data. Although there are many kinds of algorithms, sampling data have similarities, so we summarize the algorithms classification from the point of view of data characteristics and advantage.

4.2 | Classification of map-matching algorithms based on data information

As shown in Table 6, according to the data information contained in the data sets, the map-matching algorithms can be categorized into four types: geometric principle [88,100,101], topology [28,89,102,103], probability statistics [90,104–106] and advanced model [91,107–112]. These algorithms can be regarded as incremental and global method according to the number of sampling points in map matching. The incremental method first determines the previous point and then predicts the next point, while the global method takes into account the characteristics of the whole trajectory to determine the output results.

4.2.1 | Geometric principle

In general, the relationship between point-line and line-line is analyzed in the map-matching algorithms based on geometric principle, using distance, similarity and other factors to find the optimal path. In [88], by comparing the trajectory curve and the slope of the candidate segment, the least square method is used to fit the adjacent four trajectory points to the candidate segment. To improve the accuracy of real-time matching for navigation system, Teng et al. [100] proposed a new algorithm integrating ST proximity and improved weighted circle. The product of vehicle speed and travel time is used as the search radius. The elliptical region is constructed with the adjacent trajectory point as the centre, and the road in the intersecting region is the candidate segment. Because of the dynamic changing speed of vehicles, the search region also changes with

the speed. "Then, in combination with the established rules, the distance from the trajectory point to candidate segment, the angle between the driving direction and candidate segment can be determined." Zheng et al. [107] calculated the Frechet distance with discrete distance, which can determine the relationship between the trajectory curve and the road curve. Viewing distance as a factor, uncertainty inference is used in cloud model to obtain the matching similarity of road segments.

4.2.2 | Topology

The topology-matching algorithm needs to build the road network and combines the local search algorithm to find the optimal path. Zhang et al. [113] split the trajectory into segments without loops and used the stack-stored road topology to find candidate segment. Then, a custom rule is used to filter the candidate segment to obtain the matching results. Li et al. [87] took nodes, roads and intersections as objects to establish the spatial topological relationship between points and lines. They divided the matching process into four different states and implemented the corresponding matching algorithm according to each state. In [89], sensors are used to receive road network data for nearby vehicles. After using the map data of multiple vehicles to obtain the subpath, the matching position is figured out by the linear model. Because of the simulation in the ideal environment, the delay and GPS accuracy of the sensor will affect the algorithm.

4.2.3 | Probability statistics

Probability statistics algorithm is designed to find a matching path with the maximum probability. Tuo et al. [90] used candidate segment to construct map search trees, which have multiple branch paths from root to end and cannot be traced back. Then, considering the spatial and temporal information of trajectory, they evaluated each branch in the map-searching tree. Li et al. [104] proposed a probability formula to establish an elliptical region search candidate segment. According to the characteristics of roads, the correlation coefficient method is used to select the candidate segment. The deviation of the matching points is also rectified on the road segments with the largest coefficient. Cheng et al. [105] thought that there is a probability relationship between the vehicle heading angle of the trajectory point and the road segment. By applying Bayesian theorem, they designed mathematical models to obtain the highest score of candidate segment as matching results. Yao et al. [108] used the membership function to perform fuzzification of three input variables of average error, projection distance and velocity to obtain the fuzzy subset, and designed inference rules for average angle and speed. They weighted the inference results in order to obtain the highest score as the matching path.

TABLE 7   GPS sampling data

Source	Description	Span for data collection	Registration required	Download URL
The Hong Kong university of science and technology	The place of data acquisition is in Shanghai, China. Each record contains seven fields. The first field is the identification of this particular taxi. The second is the timestamp when this report is sent out. The third and fourth fields are the longitude and latitude of the current location of this taxi, respectively. The fifth is the instantaneous speed of the taxi at this moment. The sixth is the angle from the North in clockwise direction with a unit of 2 degrees. The last field is the current status of this taxi.	On 20 February 2007	None	https://www.cse.ust.hk/scrg/
Microsoft research Asia	These GPS trajectory data sets were collected in Geolife project by 182 users in a period of over 3 years. A GPS trajectory of these data sets is represented by a sequence of time-stamped points, each of which contains the information of latitude, longitude and altitude. The data sets contain 17,621 trajectories with a total distance of about 1.2 million kilometres and a total duration of 48,000+ hours.	From April 2007 to August 2012	None	https://www.microsoft.com/en-us/download/details.aspx?id=52367
Microsoft research Asia	The data sets contain the GPS trajectories of 10,357 taxis within Beijing. The total number of points in data sets is about 15 million and the total distance of the trajectories reaches to 9 million kilometres. Each line of a file has the following fields, separated by comma: taxi id, date time, longitude, latitude.	From 2 February 2008 to 8 February 2008	None	https://www.microsoft.com/en-us/research/publication/t-drive-trajectory-data-sample/
Crawdad	The data sets contain mobility traces of taxi cabs in San Francisco, USA. It contains GPS coordinates of approximately 500 taxis collected over 30 days in the San Francisco Bay area, including vehicle ID, timestamp, longitude, latitude information.	From 17 May 2008 to 10 June 2008	Applying for a username and password to download.	http://crawdad.org/epfl/mobility/20090224/
Crawdad	The data sets contain mobility traces of taxi cabs in Rome, Italy. It contains GPS coordinates of approximately 320 taxis collected over 30 days, including vehicle ID, timestamp, longitude, latitude information.	From 1 February 2014 to 2 March 2014	Applying for a username and password to download.	http://crawdad.org/roma/taxi/20140717/
Nabeel Akhtar's study	Vehicle information collected on two California highways, including vehicle longitude, latitude, time, vehicle ID and vehicle density information.	-	None	http://cs-apeople.bu.edu/nabeel/VANETs_Mobility_Dataset/
TAPASCologne	Vehicle data collected within 24 h of a working day in Cologne, Germany, including information on time, vehicle ID, latitude and longitude and speed.	One day in 2012	None	http://kolntrace.project.citi-lab.fr/#download

T A B L E 7 (Continued)

Source	Description	Span for data collection	Registration required	Download URL
Data castle	Data collected from taxis in Chengdu, China, which include vehicle ID, latitude, longitude, passenger status and time information. A reminder that the data for the 12:00 AM to 6:00 AM time period were deleted.	From 3 August 2014 to 30 August 2014	Applying for a username and password to download	https://www.pkbigdata.com/common/cmpt/

4.2.4 | Advanced model

Advanced model is relatively novel, including fuzzy logic methods, D-S evidence theory, support vector machine multi-classification matching, neural network, reinforcement learning, genetic algorithms and so on. Bi et al. [109] took the vehicle position and the direction information as the reliability parameters of the reasoning evidence, proposed two identification functions for each candidate segment, then took the maximum value of the function as the matching result. Wang et al. [91] used LIBSVM to establish multiclassification models for processed road network data and GPS sampling data. The map matching is transformed from the real position prediction problem to the multiclassification problem, in which the road network data contains label information such as road segment angle and position. These label data need to be input into the extreme learning machine algorithm, and then adjust the number of hidden layer nodes to get the output results. Sun et al. [110] proposed History Markov decision Processes Q-Learning algorithm based on reinforcement learning and historical trajectory, which include searching for candidate point set, selection of candidate path set and solution of optimal path. First of all, the candidate point set is formed by projecting to the corresponding candidate segment according to the position of the trajectory point. Then, if there is a historical track between the adjacent candidate matching points, the historical track is taken as the candidate segment. Otherwise, the shortest path of adjacent candidate points is used as the candidate path. Finally, distance, historical trajectory weight and connectivity are taken as reward functions to solve the optimal path using reinforcement learning. Gu et al. [111] proposed a map-matching method based on multicriteria genetic algorithm. They used the weighted combination of spatial similarity, shortest path distance and direction similarity as the fitness function of the genetic algorithm, and iteratively minimized the objective function to output the optimal results.

4.3 | Introduction of open source GPS sampling data sets

In this section, we introduce some open source trajectory data sets, including related descriptions, download addresses and registration requirements, as shown in Table 7. We have carried out download test for these data sets. The result is that they are all usable.

4.4 | Evaluation measurements

To estimate the system performance of the map-matching algorithms, experimental measurements are used to verify the advantages and disadvantages of the map-matching algorithms. Commonly used map-matching measurements include accuracy and time efficiency. The accuracy (e.g. correct matching percentage [CMP]) is the ratio of the number of roads correctly matched to the total number. Time efficiency (TE) is the time spent from beginning matching to outputting all matched points.

$$CMP = \frac{|Num_c|}{|Num_{all}|} \times 100\% \quad (1)$$

where Num_c is the number of correctly matched points and Num_{all} is the total number of points needed to be matched.

$$TE = T_o - T_s \quad (2)$$

where T_o represents the time of application ends, and T_s is application start-up time.

Map-matching algorithms generally need to be measured by CMP and TE measurements. In addition, some researchers also use Recall, LenCMP and Circuity to measure the performance of map-matching algorithms.

$$Recall = \frac{|Num_c|}{|Num_c + Num_l|} \times 100\% \quad (3)$$

where Num_l is the number of missing matching.

$$LenCMP = \frac{|Len_c|}{|Len_{all}|} \times 100\% \quad (4)$$

where Len_c is the length of the CMP and Len_{all} is the path length of the actual driving of the vehicle.

$$Circuity = \frac{|Len_e|}{|Len_c + Len_e|} \times 100\% \quad (5)$$

where Len_e represents the length of mismatching path.

It is worth noting that the selection of algorithmic measurements depends on whether the matching object is a trajectory point or a path length. CMP and Recall are recommended if trajectory points are selected as the study object. Otherwise, LenCMP and Circuity are often used in trajectory path measurement. No matter which one your research object is, TE always appears in the experiment as a basic metric.

In this section, we first partition map matching into sampling frequency and data information-based approaches and then introduce the open source trajectory data sets and evaluation measurements.

5 | DISCUSSIONS AND CONCLUSION

We conduct a comprehensive survey of the map-matching problem, which includes data preprocessing and selection algorithms. We first point out the shortcomings of the previous study, and then propose a new classification method for the existing measurements. In addition, we introduce the open source data sets and evaluation measurements. Finally, we present new challenges to map-matching techniques. In conclusion, this study summarizes the current trend of map-matching techniques and provides guidance for future research directions.

Map-matching technique is the basis of trajectory mining. With the development of the Internet, map-matching techniques, including data preprocessing and matching algorithms, also have new challenges. In order to further study the vehicle map-matching techniques, the following aspects can be studied in the future.

5.1 | Challenge of data preprocessing

(i) Incomplete FCD is the most common problem in data preprocessing, which will increase the difficulty of map matching. Most researchers prefer to use vehicle speed and heading angle variables for map matching, but the deletion method and adjacent value filling are inappropriate. We can analyze the overall trajectory curve and insert the missing value. If the amount of data is large enough, we can train the model with historical data, which can be used to predict the missing value. FCD with more attributes should be further explored, which is applied to map-matching algorithms in order to improve the accuracy of map matching, for example, vehicle slope, altitude and other factors; (ii) Currently, the road network structure appears mostly in the form of point line, which will inevitably lose the integrity of the road. In the three-dimensional road, high speed and other complex road, the lack of road integrity will increase the error of matching. When building a road network structure, we can add additional space to store complete roads. In addition, due to the time lag of open road data, we can cluster the massive floating vehicle data in order to restore the missing road.

5.2 | Challenge of map-matching algorithms

Although most map-matching algorithms are based on high-frequency sampling data, map-matching algorithms based on low-frequency sampling data have attracted a lot of attentions in recent years. We can focus on the following directions: (i) Due to sparsity of low-frequency data, most studies are based on hidden Markov probability or advanced model, and there is rare work on geometric principles and topological structures. We can apply the algorithms based on geometric principle and topology by combining more attributes of low-frequency sampling FCD and road data; (ii) When searching candidate segment, grid search is more popular because it works fast. When candidate segments appear in extreme positions, most people choose to continue searching for eight nearby grids to avoid missing candidate segments, but it also increases the computational overhead. It is reasonable to construct a circular region with the product of vehicle speed and driving time as radius to choose the search results of nine grids to reduce the computational overhead; (iii) The increasingly mature distributed computing platform can efficiently support large-scale data analysis and modelling problems. We can use cluster computing capabilities to improve the efficiency of map matching.

ORCID
Shaojie Qiao https://orcid.org/0000-0002-4703-780X

REFERENCES
1. Cai, M., Zhang, W., Wang, H.: Survey of data mining in Big data era. Value Eng. 38(05), 155–157 (2019)
2. Qiao, S., et al. A trajectory feature extraction approach based on spatial coding technique. Sci. Sin. Inf. 47(11), 1523–1537 (2017)
3. Qiao, S., et al. PutMode: Prediction of uncertain trajectories in moving objects databases. Appl. Intell. 33(3), 370–386 (2010)
4. Silva, A.P., Mateus, G.R.: Location-based taxi service in wireless communication environment. In: 36th Annual simulation symposium, Orlando, pp. 47–54 (2003)
5. Zhao, Y., et al. A new method of road network updating based on floating car data. In: 2011 IEEE international geoscience and remote sensing symposium, Vancouver, pp. 1878–1881 (2011)

6. Ge, Y., et al. A taxi driving fraud detection system. In: 2011 IEEE 11th International conference on data mining, Vancouver, pp. 181–190. (2011)

7. Lee, D.W., Bass, E.J., Patek, S.D.: Towards a transportation network model for air taxi dispatch planning. In: 2005 IEEE design symposium, systems and information engineering, Charlottesville, pp. 293–301 (2005)

8. Long, X., Jin, L., Joshi, J.: Exploring trajectory-driven local geographic topics in foursquare. In: Proceedings of Ubicomp '12. Pittsburgh, pp. 927–934 (2012)

9. Qiao, S., et al. A self-adaptive parameter selection trajectory prediction approach via hidden Markov models. IEEE Trans. Intell. Transport Syst. 16(1), 284–296 (2015)

10. Qiao, S., et al. Predicting long-term trajectories of connected vehicles via the prefix-projection technique. IEEE Trans. Intell. Transport Syst. 19(7), 2305–2315 (2018)

11. Dork, R.A.: Satellite navigation systems for land vehicles. IEEE Aero. Electron. Syst. Mag. 02(05), 2–5 (1987)

12. Qiao, S., et al. TraPlan: An effective three-in-one trajectory-prediction model in transportation networks. IEEE Trans. Intell. Transport. Syst. 16(3), 1188–1198 (2015)

13. Krakiwsky, E.J., Harris, C.B., Wong, R.V.C.: A Kalman filter for integrating dead reckoning, map matching and GPS positioning. In: Proceedings of IEEE PLANS'88, Orlando, pp. 39–46 (1988)

14. Pyo, J.-S., Shin, D.-H., Sung, T.-K.: Development of a map matching method using the multiple hypothesis technique. In: Proceedings of 2001 IEEE intelligent transportation systems, Oakland pp. 23–27 (2001)

15. Zhou, C., et al. Survey of map-matching algorithm for intelligent transport system. Comp. Sci. 42(10), 1–6 (2015)

16. Teng, Z., et al. Vehicle navigation map matching algorithm based on multiple-attribute integration strategy. J. Jiangsu University (Natural Science Edition). 39(114–118), 25 (2018)

17. Alakhras, M., Hussein, M.I., Oussalah, M.: Location fixing and fingerprint matching fingerprint map construction for indoor localization. J. Sens. 2020, 7801752:1–7801752:14 (2020)

18. Hong, L., et al. An INS/visual UAV autonomous landing navigation algorithm based on fuzzy prediction. Chinese J. Sens. Actuators. 2019, 1849–1855 (2019)

19. Sobreira, H.M., et al. Map-matching algorithms for robot self-localization: A comparison between perfect match, iterative closest point and normal distributions transform. J. Intell. Robotic Syst. 93(3-4), 533–546 (2019)

20. Hou, X., et al. Fast online map matching for recovering travelling routes from low-sampling GPS data. In: Proceedings of 2018 IEEE SmartWorld, Guangzhou, pp. 917–924 (2018)

21. Zhang, H.: Research on accessibility of integrated passenger transport network in Guangzhou based on ArcGIS. [Master]. Chang An University (2017)

22. Brereton, P., et al. Lessons from applying the systematic literature review process within the software engineering domain. J. Syst. Software. 80(4), 571–583 (2007)

23. Wohlin, C., Prikladniki, R.: Systematic literature reviews in software engineering. Inf. Softw. Technol. 55(6), 919–920 (2013)

24. Kitchenham, B., Brereton, P.: A systematic review of systematic review process research in software engineering. Inf. Softw. Technol. 55(12), 2049–2075 (2013)

25. AkbarV, M.A., et al. A systematic study to improve the requirements engineering process in the domain of global software development. IEEE Access. 8, 53374–53393 (2020)

26. Ghazi, A.N., et al. Survey research in software engineering: problems and mitigation strategies. IEEE Access. 7, 24703–24718 (2019)

27. Huang, M., et al. A data model for road direction. Acta Sci. Natur. Univ. Sunyatseni. 45(4), 121–123 (2006)

28. Zhang, X., et al. A map-matching algorithm for vehicle trajectory in complex road networks. Sci. Surv. Mapp. 43(08), 110–115 (2018)

29. Li, Q., Hu, B., Le, Y.: *Flowing car data map-matching based on constrained shortest path algorithm*. Geomatics and Information Science of Wuhan University, 38(07), 805–808+885 (2013)

30. Feng, J., Zhu, Z.: Modified histogram: A spatio-temporal aggregate index for moving objects in road networks. Int. J. Intell. Defence Supp. Syst. 29, 4135–4139 (2012)

31. Quddus, M.A., Ochieng, W.Y., Noland, R.B.: Current map-matching algorithms for transport applications: State-of the art and future research directions. Transport Res. C Emerg. Technol. 15(5), 312–328 (2007)

32. Hashemi, M., Karimi, H.A.: A critical review of real-time map-matching algorithms: Current issues and future directions. Comput. Environ. Urban Syst. 48, 153–165 (2014)

33. Song, Z.: A review of map matching algorithm based on FCD information acquisition. China Comp. Commun. 28(2), 82–84 (2016)

34. Chao, P., et al. A survey on map-matching algorithms. In: *Proceedings of 31st Australasian database conference on Databases theory and applications*, Melbourne, vol. 12008, pp. 121–133.Springer (2020)

35. Gao, Q., et al. Trajectory Big data: A review of key technologies in data processing. J. Softw. 28(4), 959–992 (2017)

36. Yan, F: Research on map matching algorithm based on low-sampling-rate floating car data [Master]. Huazhong University of Science and Technology (2016)

37. Huo, M.: Research on map matching method for complex road network and low frequency sampling GPS trajectory data [Master]. Chang'an University (2019)

38. Hui, F., et al. Energy consumption assessment of typical driving behavior under multiple driving conditions based on trajectory data. Comp. Appl. Softw. 37(5), 50–56 (2020)

39. Wen, J., Yang, X., Wu, C.: A characteristic analysis of driving behaviors for operation vehicles based on Big data. J. Transp. Inf. Safety. 36(4), 42–50 (2018)

40. Liu, Y., et al. Travel time prediction of main transit line based on multi-source data fusion. J. Transp. Syst. Eng. Inf. Technol. 19(4), 124–129148 (2019)

41. Zeng, X., Fan, B.: Research on matching algorithm of vehicle navigation map based on android mobile terminals. J. South China Normal Univ. (Soc Sci Ed) 47(4), 160–164 (2015)

42. Wang, J., Su, X.: Adaptive tracking test of vehicles based on GPS data in Alpine environment. Environ. Technol. 36(6), 59–64 (2018)

43. Ye, S.: Data deletion and summary of its processing methods. Electronic Test. 60(18), 65–67 (2017),

44. Dong, Y., Zhang, F.: Single car travel time estimation based on running state of low frequency floating car at intersections. J. Geomatics. 43(3), 28–32 (2018)

45. Feng, X., et al. Querying of massive trajectory data in internet of vehicles based on HBase. J. Fuzhou University (Natural Science Edition). 46(4), 466–471 (2018)

46. Tian, W., Zhang, J.: Judgment and analysis of congested road. Remote Sens. Inf. 32(1), 149–154 (2017)

47. Tan, B., Deng, Z.: Realization of the coordinate transformation method in GPS navigation system. J. Xi'an Technological University. 34(6), 504–510 (2014)

48. Zhang, Y., Zhang, B., Zhou, X.: The taxi large floating car data map matching and analysis. J. Henan Sci. Technol. 11(11), 59–62 (2016)

49. Li, C., et al. Unmanned vehicle path tracking algorithm based on differential GPS in local area. Sci. Technol. Eng. 19(25), 256–261 (2019)

50. Zhang, X., Wang, B., Fan, K.: Study on the matching between GPS and digital map. Comput. Simul. 22(6), 148–151 (2005)

51. Li, Z., et al. Research on Big data preprocessing for public transportation GPS–based on Hohhot public transportation data. J. Inn. Mong. Univ. Technol. (Soc Sci Ed). 04(37), 287–292 (2018)

52. Cai, X., Cao, Y., Dong, P.: Velocity-based trajectory stop point recognition algorithm. Comput. Syst. Appl. 29(4), 214–219 (2020)

53. Yin, L., et al. Reconstruction of Gappy GPS coordinate time series based on long short-term memory networks. J. Geomat. Technol. 35(4), 331–336 (2018)

54. Wang, F., et al. Application analysis of GPS coordinate time series interpolation based on RegEM algorithm. J. Geodesy. Geodyn. 40(1), 45–50 (2020)

55. Sheng, C., et al. Study of map-matching and interpolation algorithm of floating car data. Sci. Surv. Mapp. 44(08), 106–112 (2019)

56. Zhao, S., Qu, R., Wang, J.: A vehicle trajectory reconstruction method based on improved inverse distance weighted interpolation. J. Highw. Transp. Res. Dev. 35(10), 133–139 (2018)

57. Shi, J., Liu, C.: NGSIM vehicle trajectory reconstruction. J. Beijing Univ. Technol. 45(6), 601–609 (2019)

58. Wan, N., Vahidi, A., Luckow, A.: Reconstructing maximum likelihood trajectory of probe vehicles between sparse updates. Transport Res. C Emerg. Technol. 65, 16–30 (2016)

59. Yang, W., Ai, Y.: A method for road map construction based on trajectory segmentation and layer fusion using vehicle track line. Acta. Geod. Cartogr. Sinica. 47(12), 1650–1659 (2018)

60. Zhang, T., Yang, Z.: Segmentation-based trajectory simplification algorithm. Appl. Res. Comput. 36(7), 2044–2048 (2019)

61. Zhang, Y.: Clustering algorithm based on trajectory segment moving objects segmentation. Comput. Digi. Eng. 42(1), 69–72 (2014)

62. Kang, J., et al. Parallelized map matching algorithm for large scale trajectory data. Meas. Control Technol. 38(02), 98–102 (2019)

63. Xia, Y., et al. Grid-based k-nearest neighbors query over moving trajectories under spark. J. Chongqing Univ. Posts Telecommun. (Nat. Sci. Ed.) 31(4), 531–537 (2019)

64. Chen Z., et al. Top-k trajectory query based on traveling time. J. Chinese Comput. Syst. 40(7), 1496–1502 (2019)

65. Hong, Z.: Using Mapinfo data to build spatial index based on MAPX. Geomat. Spatial Inform. Technol. 36(3), 118–121,124,128 (2013).

66. Wang, T., Shen, Z.: Research on passengers hotspot of taxi technology. J. Harbin Univ. Commerce (Nat. Sci. Ed.) 35(5), 590–594,606 (2019)

67. Ruan, S., et al. Vehicle trajectory extraction algorithm based on license plate recognition data. J. Zhejiang Univ. (Eng. Sci.). 52(5), 836–844 (2018)

68. Zhong, H., et al. Algorithm of bus route trajectory based on GIS road network. Comput. Syst. Appl. 26(11), 182–186 (2017)

69. Luo, Y., Wei, B., Fu, Y.: The construction method of hierarchical road network topological structure based on MapX. Appl. Sci. Technol. 39(05), 32–35 (2012)

70. Zhang, Y., et al. Analysis and implementation of shortest path algorithm based on OpenStreetMap. Comput. Technol. Dev. 23(11), 37–41 (2013)

71. Yu, X., Han, B., Zhai, J.: Research on structural discrimination and classification between dendritic network and grid-like network. Forest Eng. 36(178–86), 95 (2020)

72. Huang, L., Jia, J., Hou, A.: Research on key technologies about automatic differential fusion of navigation road network data. Geospatial Inform. 18(2), 21–23 (2020)

73. Zhang, Y., et al. Analysis and implementation of shortest path algorithm based on OpenStreetMap. Comp. Technol. Develop. (11), 37–41 (2013)

74. Sun, D., et al. Algorithm based pre-processing for constructing topological structure of urban road network. Comput. Eng. Appl. 44(23), 233–235,248 (2008)

76. Zeng, Y., et al. Structure fractal analysis of road network on multi-geospatial scales. Geomat. World. 25(5), 35–40 (2018)

77. Deng, M., et al. Fine road network modeling based on ubiquitous location data. J. Cent. S. Univ. 50(9), 2171–2183 (2019)

78. Fan, N., et al. Map matching algorithm for urban road network based on dynamic weight. Meas. Control Technol. 37(1), 154–158 (2018)

79. Wang, Y., Gu, W., Feng, L.: A map matching algorithm based on subfield search method. J. Shandong Instit. Light Ind. (Nat. Sci. Ed.). 29(4), 77–80 (2015)

80. Li, D., et al. The design and implementation of a map matching algorithm. Navig. Position. Timing. 04(02), 31–34 (2017)

81. Zhao, S., Zhang, J., Qu, R.: An improved map matching algorithm for floating car. Bull. Surv. Mapp. 64(1), 97–102 (2018)

82. Liu, S., et al. Parallel raster image tile algorithm based on MPI. Comput. Eng. Appl. 54(1), 48–53111 (2018)

83. Wang, M., Cheng, L.: Study on map-matching algorithm for floating car. Acta. Geod. Cartogr. Sinica. 41(1), 133–138 (2012)

84. Liu, W., Wang, J., Zheng, L.: Map-matching algorithm research based on spatio-temporal analysis. Appl. Res. Comput. 33(8), 2266–2269 (2016)

85. Sun, H., et al. Excessive-emission vehicles real-time track matching algorithm based on road network topology and weights. In: Proceedings of the 2019 Chinese Control Conference. Guangzhou, pp. 6325–6332 (2019)

86. Chen, Y., Wu, C., Yao, M.: Dynamic topology construction for road network. In: Proceedings of the 4th International conference on networked computing and advanced information management. Gyeongju, pp. 359–364 (2008)

87. Li, X., et al. Vector road matching algorithm based on a topological path. J. Image Graph. 22(05), 596–609 (2017)

88. An, L., et al. Map matching algorithm based on course vertical line. Sci. Surv. Mapp. 42(06), 112–116 (2017)

89. Choi, S.T., Hur, W.S., Seo, S.W.: Cooperative localization based on topology matching. In: Proceedings of the 2014 IEEE 6th international symposium on wireless vehicular communications. Vancouver, pp. 1–5 (2014)

90. Tuo, Y., Yun, X., Zhang, Y.: MSTMA novel map matching approach for low-sampling-rate trajectories. In: Proceedings of the 2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC). Montreal, pp. 1–7 (2017)

91. Wang, L.: Research on map matching of the vehicle tracks based on multi-classification algorithms. Intell. Comput. Appl. 07(06), 35–38 (2017)

92. Yuan, J., et al. An interactive-voting based map matching algorithm. In: Proceedings of the 11th international conference on mobile data management. Kanas City, pp. 43–52 (2010)

93. Gao, W., Li, G., Ta, N.: Survey of map matching algorithms. J. Softw. 29(02), 225–250 (2018)

94. Zeng, J., Sun, L., Wang, X.: Research on GPS track data map matching algorithm of Beijing. Beijing Surveying and Mapping. 33(03), 255–260 (2019).

95. Chen, B., et al. An integrated map-matching algorithm based on GPS. Electron. Sci. Technol. 27(12), 20–23 (2014)

96. Liu, Y.: One new map matching model for vehicle-borne navigation system. In: Proceedings of the 2010 international conference on optoelectronics and image processing. Haikou, pp. 66–68 (2010)

97. Zhang, Y., et al. A multi-weights map matching algorithm used in GPS system for vehicle navigation application. In: Proceedings of the 2014 12th international conference on signal processing (ICSP). Hangzhou, pp. 2375–2378 (2014)

98. Liu, M., et al. An improved map matching algorithm based on HMM model. Acta Sci. Natur. Univ. Pekinensis. 54(06), 1235–1241 (2018)

99. Lou, Y., et al. Map-matching for low-sampling-rate GPS trajectories. In: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. Seattle, pp. 352–361 (2009)

100. Teng, W., Wang, Y.: Real-time map matching: A new algorithm integrating spatio-temporal proximity and improved weighted circle. Sci. Lett. 11, 288−297 (2019)

101. Taylor, G.E., et al. Road reduction filtering for GPS-GIS navigation. Trans. GIS. 05(03), 193–207 (2001)

102. Liu, M., Li, M.:: Research on floating car map matching algorithm. In: Proceedings of the 2017 25th International Conference on Geoinformatics. Buffalo, pp. 1–5 (2017)

103. Liu, Z., Wang, X., Yan, X.: Map-matching algorithm for GPS trajectories in complex urban road networks. J. Univ. Electron. Sci. Technol. China. 45(06), 1008–1013 (2016)

104. Li, X., et al. Map matching algorithm design based on GPS information. In: Proceedings of the 2014 IEEE Workshop on Electronics, Computer and Applications, Ottawa, pp. 898–901 (2014).

105. Cheng, Y., et al. A quick map-matching algorithm based on spatio-temporal Bayesian model. Commun. Sci. Technol. Heilongjiang. 41(09), 175–177+181 (2018).

106. Liu, X., et al. A ST-CRF map-matching method for low-frequency floating car data. IEEE Trans. Intell. Transport Syst. 18(05), 1241–1254 (2017)

107. Zheng, S., et al. A map matching algorithm based on discrete Frechet distance. J. Hefei Univ. Technol. (Nat Sci). 40(01), 42–46 (2017)

108. Yao, E., Zuo, T.: Real-time map matching algorithm based on low-sampling-rate probe vehicle data. J. Beijing Univ. Technol. 39(06), 909–913 (2013)

109. Bi, J., Fu, M., Zhang, Y.: A map matching algorithm for vehicle navigation systems based on dead reckoning (D-S) evidence reasoning. Trans. Beijing Institute of Technology. 22(3), 393–396 (2002)

110. Sun, W., Xiong, T.: A low-sampling-rate trajectory matching algorithm in combination of history trajectory and reinforcement learning. Acta Geod. Cartogr. Sinica. 45(11), 1328–1334 (2016)

111. Gu, Y., Lu, W., Shao, Z.: Map-matching method based on multi-criteria genetic algorithm and low-sampling-rate floating car data. Acta Geod. Cartogr. Sinica. 45(06), 585–592 (2019)

112. Gong, Y.J., et al. Ant mapper: An ant colony-based map matching approach for trajectory-based applications. IEEE Trans. Intell. Transport Syst. 19(02), 390–401 (2018)

113. Zeng, Z., et al. Curvature integration constrained map matching method for GPS floating car data. Acta Geod. et Cartogr. Sin. 44(10), 1167–1176 (2015)