Anomalous Taxi Trajectory Detection using Popular Routes in Different Traffic Periods

Lina Xu, Yonglong Luo*, Qingying Yu, Xiao Zhang, Wen Zhang, Zhonghao Lu
School of Computer and Information, Anhui Normal University, Wuhu 241002, Anhui, China
Anhui Provincial Key Laboratory of Network and Information Security, Wuhu 241002, Anhui, China

***Abstract*—Anomalous trajectory detection is an important approach to detecting taxi fraud behaviors in urban traffic systems. The existing methods usually ignore the integration of the trajectory access location with the time and trajectory structure, which incorrectly detects normal trajectories that bypass the congested road as anomalies and ignores circuitous travel of trajectories. Therefore, this study proposes an anomalous trajectory detection algorithm using the popular routes in different traffic periods to solve this problem. First, to obtain popular routes in different time periods, this study divides the time according to the time distribution of the traffic trajectories. Second, the spatiotemporal frequency values of the nodes are obtained by combining the trajectory point moments and time span to exclude the interference of the temporal anomaly trajectory on the frequency. Finally, a gridded distance measurement method is designed to quantitatively measure the anomaly between the trajectory and the popular routes by combining the trajectory position and trajectory structure. Extensive experiments are conducted on real taxi trajectory datasets; the results show that the proposed method can effectively detect anomalous trajectories. Compared to the baseline algorithms, the proposed algorithm has a shorter running time and a significant improvement in *F-Score*, with the highest improvement rate of 7.9%, 5.6%, and 10.7%, respectively.**

***Keywords*—Anomalous trajectory detection; time periods; popular routes; gridded distance**

I. Introduction

With the rapid development of GPS positioning and wireless communication technology, more and more trajectory data have been collected [1, 2]. Detecting anomalous trajectories from a large amount of collected trajectory data has been widely used in various fields, such as fraud detection [3], medical treatment [4], and anomalous trajectory detection [5]. The huge trajectory data generated by vehicle location acquisition also provides an unprecedented opportunity to analyze the anomalies of moving objects and discover some basic rules of their movements [6]. Through timely and effective analysis of these traffic trajectories, the behavior rules of vehicles can be detected, thereby revealing the dynamic changes of certain behaviors and the “special” events hidden behind the vehicle behavior patterns [7]. For example, detours can be detected during taxi driving [8], and traffic jams due to traffic accidents [9] or temporary road closures [10]. This information can better guide the driving of taxi routes and provide early warnings of potential safety hazards, thus improving urban traffic planning.

Anomalous trajectory detection is one of the hot research topics in trajectory pattern mining [11, 12]. Usually, an anomaly means that a data object has a large deviation from the remaining objects in the retrieved dataset due to some of its unusual characteristics. For these different anomalous trajectories, many anomaly detection methods have been proposed. The existing anomalous trajectory detection technology mainly uses the similarity measurement method to find out the trajectory that is highly different from others [13-15]. The LDTRAOD [16], DB-TOD [17], and the ATD-outliers method [18] combine attributes such as distance between trajectory points, local density, and trajectory shape to calculate the abnormal score of trajectories by quantifying the similarity between trajectories. However, the time complexity of these methods is too high, which leads to prohibitive costs. Therefore, trajectory grid processing is often exploited to solve this problem, reducing the computational complexity by mapping the spatial locations of trajectory points onto the grid codes. In addition, it can be seen from the existing methods that the key to anomalous trajectory detection is extracting a representative feature of each trajectory and then using a function based on this to measure the similarity between them. Therefore, a taxi driving a long distance or a trajectory with fewer same driving trends is considered to be an anomalous trajectory. Although this can detect a large number of anomalies, the false positives of the detection results are usually high. Because these methods focus only on the location of the trajectory and do not integrate it with the time period and the trajectory structure, thus incorrectly detect normal driving trajectories of particular time periods as anomalies, and ignore loop driving behavior on normal routes.

For example, a passenger Alice takes a taxi from the starting point S to the destination point D. There are two routes at fixed starting and ending points, namely R and R_0, where the distance of R_0 is shorter than that of R. If the trip occurs during the off-peak period, which means that the traffic volume is relatively low during this period. Therefore, considering the time and fare spent by passengers on the trip, the route R_0 is chosen as the optimal route for this road segment. However, if the trip occurs between 7:00 am-9:00 am, the route R_0 may cause serious road congestion and traffic accidents due to the traffic flow during rush hour. Therefore, in terms of the time peak, the optimal choice for taxi drivers should be R.

Obviously, as the dynamics of popular routes change, the anomaly judgment of the trajectory also changes. The driver's choice of route usually changes at special periods or moments when special events occur. Therefore, it is necessary to divide the effective time intervals according to the vehicle travel pattern. However, if the division is performed at very short and fixed intervals, it will consume a lot of unnecessary time and space resources, and the detection results will also contain certain errors. Research [13] pointed out that the taxi trajectory tends to take more time during peak periods than during off-peak periods. Inspired by this, it can be determined that drivers will choose different routes at specific time periods to save passengers' travel time. Therefore, this paper proposes an anomalous trajectory detection method using popular routes in different traffic periods (PRTP). Compared with previous detection algorithms that only consider the location of trajectory points or have large time consumption, the PRTP algorithm divides the time of day into four time periods according to the temporal distribution density of trajectories, thus discovering the dynamic changes of the popular routes and considering the effect of time segmentation and trajectory time span on frequency. Meanwhile, the location and structure of the trajectory are considered comprehensively in the anomaly judgment, which improves the stability and accuracy of detection.

In summary, the main contributions of this study are as follows.

- An anomalous trajectory detection method using popular routes in different time periods is proposed, which divides the time according to the traffic flow and combines the location and time features of trajectories in different time periods to detect anomalies, thus improving the accuracy of anomaly detection.
- By integrating the access location, time, and time span of the trajectory, a calculation method of spatiotemporal frequency is designed, so that the popular routes can be accurately obtained without the interference of temporal anomalous trajectory.
- A grid distance formula is proposed to quantitatively measure the trajectory anomaly distance, which considers both the location and structure of the trajectory, enabling the detection of circuitous driving anomalies of the trajectory in addition to location anomalies. The effectiveness and performance of the PRTP method are empirically evaluated using real taxi trajectories

The rest of the paper is organized as follows. The related work on anomalous trajectory detection is reviewed in Section II. After the problem definitions are given in Section III, the main steps of the PRTP are presented in Section IV. In Section V, the overall experimental setup and experimental results are reported. Section VI discusses the proposed method. Finally, the conclusions are presented in Section VII.

II. Related Work

At present, the field of anomalous trajectory detection has been developed rapidly. Though the existing anomalous trajectory detection algorithm has been well applied in real life, further explorations are still needed. The existing detection methods can be mainly divided into two categories, anomaly detection based on spatial attributes and anomaly detection based on spatiotemporal features.

A. Detection Based on Spatial Features

Among the existing anomaly detection algorithms, the algorithms focusing on the spatial position of the trajectory occupy a larger proportion. These methods focused on considering the spatial attributes of trajectory points, such as position, direction, velocity, etc., and select specific attributes as trajectory similarity criteria to detect anomalous trajectories. Lee et al. [19] proposed a trajectory anomaly detection algorithm TRAOD, which includes trajectory division and trajectory detection. All trajectories are divided into t-partitions according to a novel partition-and-detect framework, and then trajectory partitions are detected by a distance-based method. To detect anomalous from the locally dense trajectory, Luan et al. [16] proposed the anomaly detection algorithm LDTRAOD based on the local density of trajectories, which uses the partition-and-detect framework [19] to determine local anomalous trajectories by calculating the local density and local outlier factors for each t-partition. Although these methods can quantify the similarity between trajectories, anomaly detection based on individual location attribute does not apply to anomalies with complex distributions. To combine multiple features for anomaly detection, San Roman et al. [20] proposed the CaD method which considers the angle difference, Euclidean distance, and the number of points in each trajectory, and effectively detects anomalous trajectories in each cluster by using the unsupervised learning method of the trajectory distance matrix. Sun et al. [21] proposed the TODDT algorithm in combination with dynamic difference threshold to detect anomalous trajectories from multiple perspectives. Considering the trajectory common slice subsequence, Yu et al. [22] proposed a novel trajectory anomaly detection algorithm based on common slice subsequence (TODCSS), which combines features such as direction, position, and continuity to achieve more accurate trajectory anomaly detection. And to detect different patterns of anomalous trajectories, Wang et al [23] proposed a DIS metric. After gridding the trajectories, the DIS values are used to quantify the similarity between trajectories and then assign the anomalous trajectories to different classifications. Considering the special conditions of anomalous trajectories and the motion patterns mined in the massive trajectory data, researchers have also proposed many methods. Inspired by the "few and different" property of anomalous trajectories, Zhang et al. [24] proposed the iBAT algorithm which can effectively detect outliers of trajectories by isolating the number of steps used by the trajectory. Combine the actual inherent travel flow and other information of trajectory data to make abnormal judgment. Yu et al. [25] proposed a multi-level approach that combines association rule techniques to discover moving routes. To mine the special driving patterns of moving objects from trajectories, Zhang et al. [26] proposed an algorithm to discover popular routes from fixed locations. The algorithm meshes the region area into a regular grid to discretize the historical trajectory and finally determines the most popular routes by an ant colony optimization method. Tang et al [27] used the theory of time geography to propose a path-oriented traffic state estimation model to find the most likely road path, link travel time and activity duration at possible intermediate stations.

Real-time performance is the primary requirement of practical applications when detecting trajectory anomalies. Traffic conditions vary at different times and in different locations, exploiting latent patterns in real time can capture anomalous behaviors in real-time and make timely traffic adjustments [28]. Therefore, while overcoming the drawbacks brought by the single spatial attribute algorithm, researchers have investigated anomaly detection for evolutionary trajectories. To reduce the problem of the high false alarm rate of anomaly detection due to complex road conditions, Chen et al. [29] proposed an online anomaly trajectory based on multidimensional criteria for real-time anomaly detection by simultaneously considering multidimensional criteria such as similarity, time, and distance. Ge et al. [30] proposed an evolutionary trajectory outlier detection method TOP-EYE, which continuously calculates the outlier score for each trajectory in a cumulative manner and proposed a decay function to mitigate the impact of historical trajectories on the outlier score of trajectory evolution, thus identifying evolutionary anomalous trajectories at an earlier stage. Using a street-based trajectory delineation approach, Shi et al. [31] proposed RUTOD, an anomaly detection framework for real-time urban traffic. The framework combines an individual anomalous moving object with the group anomalous values generated by various moving objects for anomalous detection.

B. Detection Based on Spatiotemporal Features

In the practical application of anomalous trajectory detection, in addition to various situations where spatial attributes need to be considered, temporal features are also important for anomalous trajectory detection. In recent years, a lot of research and exploration has been conducted on time attributes. Zhu et al. [32] proposed a new time-dependent popular route algorithm TPRO. In this algorithm, the abnormal trajectory is dynamically measured by considering the spatial and temporal characteristics of the motion trajectory. However, the TPRO algorithm is limited to the detection of historical trajectories, and it is not applicable to anomalies of real-time trajectories. Therefore, in the TPRRO [33], a real-time outlier detection algorithm was proposed based on time-dependent popular routes, which can realize efficient detection and evaluation of the testing trajectory through offline preprocessing and online anomaly detection.

Besides, for the detection of more complex anomalous trajectories in the spatiotemporal dimension, Yang et al. [34] proposed a new trajectory clustering algorithm TAD, which is based on the spatiotemporal density analysis of trajectory data to detect the stay of complex trajectories. To detect behaviors such as abnormal parking and stranding of vehicles in the time dimension, He et al. [35] proposed a common subsequence-based spatiotemporal anomaly trajectory detection method (STADCS). This method detects spatiotemporal anomalies by combining spatial features and temporal dimensions of trajectory subsequences. And based on the travel time of the trip, Eldawy et al. [36] proposed a novel real-time anomalous trajectory detection system called FraudMove. When choosing the best route, the FraudMove method takes the popular route as the best choice. Then, an adjustable time window parameter is used to control the detection times of anomaly detection, thus dynamically detecting anomalous taxi behaviors in combination with travel time.

Although most of the above anomalous detection methods combine multiple trajectory attributes to achieve different modes of anomalous trajectory detection, there are still two main limitations. First, if the spatial features or temporal features of trajectories are used alone, such detection results will produce high false positives. Researchers only consider the matching degree of trajectory points when considering the node frequencies in different time periods, the detection accuracy will be reduced by ignoring the influence of other attributes on the node frequencies. Second, most algorithms only consider the location abnormality of trajectories and ignore the change of trajectory structure in the abnormality judgment. Therefore, it is necessary to divide the time periods in anomalous trajectory detection and calculate the frequency in the spatiotemporal dimension, and integrate the trajectory structure with the trajectory position to improve the accuracy of abnormal driving trajectory detection.

III. Problem Definition

This section defines some related terms and provides formal definitions of the problems considered in this paper to facilitate further descriptions.

Definition 1 (Point): given a record, let *x*, *y*, and *t* be longitude, latitude, and timestamp, respectively. Then the spatiotemporal information can be recorded in a triplet, namely (*x*, *y*, *t*), which is a spatiotemporal point formed by the object passing through the position (*x*, *y*) at time *t*.

Definition 2 (Trajectory): a trajectory is denoted by *T*, which consists of a series of trajectory points:

$$T_i = \{p_1, p_2, \dots, p_j, \dots, p_k, \dots, p_{len}\}, \tag{1}$$

the entire trajectory dataset is represented as:

$$TS = \{T_1, T_2, T_3, \dots, T_n\}, \tag{2}$$

$|TS|$ is the size of *TS*, i.e., $|TS|=n$. p_j is the *j*-th point of T_i, which includes three components *x*, *y*, *t*. The trajectory segment is the line segment between p_j and p_{j+1}.

Definition 3 (Road Network): the road network is represented as a directed graph $G(V,E)$, where V represents the set of nodes (such as the start and end points of a road segment), and E is the set of edges (such as a road segment). In this paper, v_i is used to represent a specific vertex in *G*. If v_i and v_j are the two points of the edge *e*, it can be expressed as $e.v_i$ and $e.v_j$.

Definition 4 (Mapped Trajectory): A mapped trajectory *MT* is generated by the function $\phi(T)$, which maps latitude and longitude coordinates to the cell, and generates a series of grid codes $c_1, c_2, c_3, \dots, c_{len}$, where $c_i = \phi(p_i)$.

Definition 5 (Equivalent Cell): Given a grid size of $\mathcal{M}$, the road network is divided into equal cells of size $\mathcal{M}$, and the position of cell is denoted by $\mathbb{C}$. Then, let trajectories pass through the road network regions, if two different trajectory points p_i and p_j fall into the same cell, they are called points with the equivalent cell, which is expressed as $Equcell_{\mathcal{M}}(p_i, p_j)$.

As shown in Fig. 1, there are two trajectories and all their trajectory points have corresponding grid cells. The points p_3' and p_5 on the different trajectories fall into the same grid cell (green grid), so they are called points with the equivalent cell.

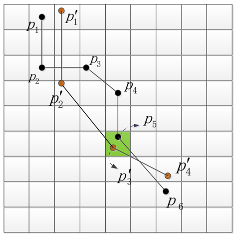


Fig. 1. An illustration of the equivalent cell.

Definition 6 (Time Span): Given a trajectory T_i, the travel time from the source to the destination is defined as the time span, which is denoted by St_i as:

$$St_i = T_i.t_d - T_i.t_s, \tag{3}$$

where $T_i.t_s$, $T_i.t_d$ represent the start time and end time of the trajectory T_i, respectively.

Definition 7 (Route): After matching the trajectory with the city map using ArcGIS, the route is represented by each node cell and its frequency. A route based on cell frequency is represented as:

$$R = \{rc_1, rc_2, rc_3, \dots, rc_m\}, \tag{4}$$

where rc_i=(c_i, f_i), c_i represents the i-th node cell on the route, f_i refers to the frequency of passing through the node cell.

Definition 8 (Anomaly trajectory): The trajectory that deviates significantly from the popular routes in the access location or structure is defined as an anomaly trajectory.

As shown in Fig. 2, there are a few trajectories between *S* and *D*, which can be divided into two groups: normal trajectories and abnormal trajectories. T_1 and T_3 are considered abnormal because they deviate from the regular routes at a certain time period.

Problem statement: Given the trajectory dataset $TS = \{T_1, T_2, T_3, \dots, T_n\}$, this study aims to detect anomaly trajectories in TS, for all $T_i \in TS$, if T_i deviates from its corresponding popular routes or drive in a roundabout manner, it is considered an anomaly trajectory.

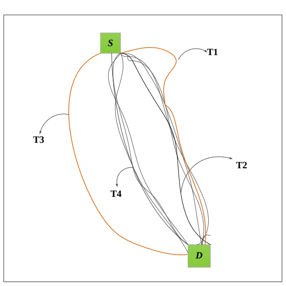


Fig. 2. An illustration of trajectories between *S* and *D*.

IV. PRTP

This section provides an overview of the anomalous trajectory detection framework proposed in this study. The PRTP method consists of two stages: trajectory preprocessing and anomalous trajectory detection. The popular routes acquisition and anomaly judgment constitute the trajectory anomaly detection stage. Fig. 3 shows the working mode of the algorithm.

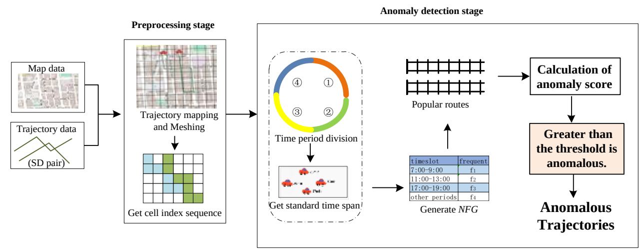


Fig. 3. Overview of PRTP approach.

A. Preprocessing Taxi Trajectory

Since anomalous driving of taxis usually occurs when they are carrying passengers, the first task in the trajectory processing stage of this study is to classify the trajectories according to their occupancy status. As shown in Fig. 4, the figure shows the trajectory of a taxi in a region for one month, in which the red line segment is the trajectory of a taxi carrying passengers and the blue line segment is the trajectory of a taxi without passengers. Therefore, for a more meaningful study, the trajectory represented by the red line segment is extracted as a valid trajectory during the preprocessing of the trajectory data in this paper.

After extracting the valid trajectories from the original trajectories, these trajectories usually contain different starting and ending points, in which there may be a certain starting and ending point between which there are not enough trajectories to form a normal trajectory group, thus causing interference to the anomaly detection. Therefore, in this study, all taxi trajectories that pass through the same *SD* pair are grouped to form the *SD* trajectory dataset (trajectories with the same source and destination). The *S* and *D* regions are 1000m×1000m grid-cells, that is, the source and destination points are satisfy $Equcell_{1000}(MT_{1.s},\dots,MT_{n.s})$, $Equcell_{1000}(MT_{1.d},\dots,MT_{n.d})$.Then , the city map is divided into grid cells of the same size, and the extracted valid *SD* trajectories are mapped into the grid cells, thus forming a mapped trajectory consisting of a series of grid cell sequences.

Fig. 4. Trajectories of a taxi in San Francisco during a month.

B. Anomaly Detection

1) Acquisition of popular routes: By extracting a specific SD trajectory dataset from the original dataset and mapping it to the corresponding urban road network, all the routes traversed by the trajectory dataset are obtained. Traversing the trajectory data and calculating the node frequency based on its trajectory point location and time, thus obtaining the node frequency graph (NFG) as shown in Fig. 5. In Fig. 5, V_s represents the source address, V_d represents the destination address, and the table corresponding to each node indicates the frequency of that node in each time period. For the division of time periods, the study divides the whole time domain into four time periods according to the peak traffic flow period, namely the morning peak period (7:00 am-9:00 am), the afternoon peak period (11:00 am -13:00 pm), the evening peak period (17:00 pm -19:00 pm), and the time period consisting of the remaining hours, and then performs anomaly detection according to testing trajectory and the popular routes in its time period.

Definition 9 (Equivalent time periods): Given two time periods t_1- t_2 and t_3- t_4, if t_1,t_2,t_3, and t_4 are all in the same pre-divided time period, they are called equivalent time periods.

According to the above analysis, if there are time periods 7:30 am-8:15 am and 8:45 am-9:00 am, because their starting time and ending time are included in the pre-divided morning peak time period, these two time periods are referred to as equivalent time periods in the study. And the trajectories of the equivalent time period have the same popular routes.

The anomalous trajectories are few and different, so the routes chosen by a large number of vehicles are often correct. Therefore, researchers usually determine the popularity of routes in each time period based on the number of times the route is visited. However, when there is a route with a large number of trajectories passing through, if only the number of trajectories is considered, the route must be identified as a popular route. But if all the trajectories on it have time anomalies, then this route is not desirable. So when calculating the frequency values, the study not only considers whether the trajectory points coincide with the nodes but also judges the time anomaly of the trajectories. Therefore, to exclude the influence of time-anomalous trajectories in calculating the frequency of route nodes, this study designs a calculation method for spatiotemporal frequency.

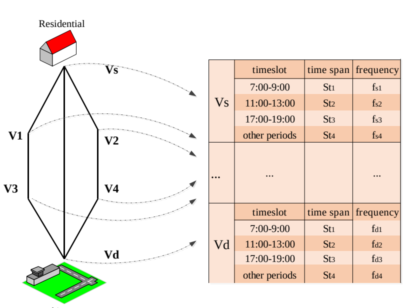


Fig. 5. An illustration of NFG.

By comparing the trajectory time span with the standard time span, the frequency coefficient of the trajectory is obtained, which is calculated as:

$$fc_i = \begin{cases} 1 - \frac{St_i - \overline{ST_j}}{\overline{ST_j}}, & \text{if } St_i > \overline{ST_j} \text{ and } St_i - \overline{ST_j} < \overline{ST_j} \\ 1, & St_i \leq \overline{ST_j} \\ 0, & else \end{cases} \quad (5)$$

where St_i represents the time span of the *i*-th trajectory, $\overline{ST_j}$ represents the standard time span of the *j*-th time period, and the value of *j* ranges from 1 to 4. The standard time span of each time period is obtained by selecting half of the normal trajectories in that time period and calculating the average time span of these trajectories.

After calculating the frequency coefficient of each trajectory passing through the node, the node frequency is defined as:

$$f_v = \sum_{i=1}^{N} fc_i \,, \quad (6)$$

where *N* is the number of trajectories passing through the node, the frequency of node *v* is the sum of the frequencies of *N* trajectories.

To obtain the popular routes of the testing trajectory, the popularity of each route needs to be compared. From the above analysis, the routes popularity can be evaluated by the node frequency values. Therefore, for route popularity comparison, the routes are expressed in the following form, i.e., R=($rc_1.f_1$,$rc_2.f_2$, ... ,$rc_m.f_m$), R'=($rc_1.f'_1$,$rc_2.f'_2$, ... ,$rc_n.f'_n$) . When $rc_x.f_x = rc_x.f'_x$ ($x \in \{1,2, \dots, p\text{-}1\}$, where $p>2$), if $rc_p.f_p > rc_p.f'_p$, then the route R is more popular than route R', denoted by $\mathcal{P}_{\mathrm{R}} > \mathcal{P}_{\mathrm{R}'}$.

After the analysis of route popularity, the first *k* high-frequency routes of the testing trajectory are called popular routes (*PR*), that is, if there is a route set *RS*= $\{R_1, R_2, \dots, R_k, \dots, R_q\}$, where $\mathcal{P}_{R_1} > \mathcal{P}_{R_2} > \cdots > \mathcal{P}_{R_k} > \cdots > \mathcal{P}_{R_q}$, then the popular routes *PR*=$\{R_1, R_2, \dots, R_k\}$.

Algorithm 1 Obtain the Popular Routes of the trajectory

Input: Road network *G*, trajectory dataset *TS*, Grid size $\mathcal{M}$, number of popular route *k*.

Output: Popular routes *PR*.

```
gridLabel ← Mesh the road network in ℳ size
Map the TS to gridLabel to get RS={R_1, R_2, R_3, ..., R_q}
TS is divided into four groups according to the time period of trajectory
for i in RS do
   for j in TS do
      Select trajectory group according to the time period of MT_j
      Calculate St_j, fc_i according to equations (3), (5), respectively
      Calculate f of R_i according to equation (6)
   end for
end for
PR← select the k-popularity routes
return PR
```

The overall pseudo-code for obtaining the popular routes of the trajectory is given in Algorithm 1. First, the trajectories are mapped to the gridded road network to obtain the RS through which the trajectory dataset passes (lines 1-2). Then, the dataset is partitioned and trajectories belonging to the same time period are grouped (line 3). Finally, each trajectory is traversed to find the frequency values of the routes in different time periods, and the popular routes are obtained based on the popularity comparison (lines 4-11). The time complexity of Algorithm 1 is $O(n \cdot q)$, where *n* is the number of trajectories and *q* is the number of routes.

2) Perform trajectory anomaly judgment: After the popular routes of the testing trajectory are obtained according to the above steps, the PRTP algorithm performs trajectory anomaly detection by comparing the testing trajectory with the top-*k* popular routes.

In this study, position anomalies and structural anomalies of trajectory are mainly considered in calculating trajectory distance. The position anomaly focuses on whether the visited positions are consistent, and the structure anomaly focuses on the circuitous intersection of trajectories. Before obtaining the trajectory distance, the trajectory and popular route are converted into position access matrixes according to their access position. The elements of the *LocMatrix* are as:

$$LocMatrix_{MT_i}(\mathbb{C}) = \begin{cases} 1, & \text{if } \mathbb{C} \text{ is accessed by } MT_i \\ 0, & \text{else} \end{cases}, \quad (7)$$

where $\mathbb{C}$ denotes the position in matrix.

The structure matrix $StrMatrix_{MT_i}$ is obtained by determining the circuitous access grid through the number of trajectory visits. The elements of the $StrMatrix$ are as:

$$StrMatrix_{MT_i}(\mathbb{C}) = \begin{cases} 1, & \text{if } Num.\mathbb{C} > 1 \\ 0, & \text{else} \end{cases}, \quad (8)$$

where $\mathbb{C}$ denotes the position in *StrMatrix*, $Num.\mathbb{C}$ denotes the number of times the trajectory MT_i visits position $\mathbb{C}$.

In the study, the number of common locations and circuitous visits are focused on, and then define the distance metric of the trajectory with the matrix information. Given the gridded trajectory MT_i and one of the corresponding optimal routes PR_j , transform them into $LocMatrix_{MT_i}$ and $LocMatrix_{PR_j}$, respectively, and determine the $StrMatrix_{MT_i}$ according to the number of trajectory visits. Then the anomaly distance is calculated according to the following defined formulas:

$$LocMatrix_{common} = LocMatrix_{MT_i} \cap LocMatrix_{PR_j}, \quad (9)$$

$$GDDis(MT_i, PR_j) = \frac{|LocMatrix_{MT_i} - LocMatrix_{common}| + |StrMatrix_{MT_i}|}{|LocMatrix_{MT_i}|}, \quad (10)$$

where $|*|$ denotes the summation over the elements of the matrix $*$, $LocMatrix_{common}$ is the matrix of the common position of the testing trajectory access matrix and the popular route access matrix.

Meanwhile, from equation (10), it can be seen that the higher the matching degree between the testing trajectory and popular routes, the greater the value of $|LocMatrix_{common}|$, and the closer the *GDDis* distance is to 0. Therefore, the specific value of the *GDDis* distance can be used to quantify the match between the testing trajectory and popular routes.

In Fig. 6, given the trajectory MT_1 and the route PR_1, three 8×8-sized matrices can be obtained. From the matrix calculation, $|LocMatrix_{MT_1}|$ =19, $|LocMatrix_{PR_1}|$ =12, the intersection of the matrices $|LocMatrix_{common}|$=11, and the $|StrMatrix_{MT_1}|$ =7. Therefore, the $|LocMatrix_{MT_i} - LocMatrix_{common}|$ =8, the number of abnormal cells is 8+7=15, then, the *GDDis* of MT_1 is 15/19=0.79. However, if MT_1 only considers the anomaly of the access location, the final *GDDis* is 8/19 = 0.42, which may misjudge MT_1 as a normal trajectory. Therefore, through the different results of the above two calculations of MT_1, it can be seen that the study combines structural anomalies with location anomalies improves the detection accuracy of anomalous trajectories.

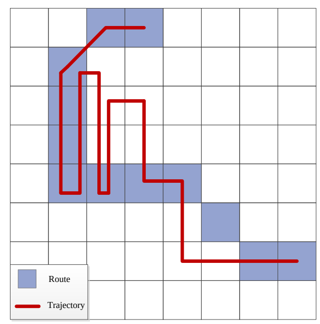


Fig. 6. An access schematic of trajectory and route.

Based on the study [32], there are often multiple alternative routes between two locations, and not all of them have the same popularity. Therefore, when judging the trajectory anomaly, the popular routes should occupy different weights in the anomaly score. Suppose exist popular routes $PRS=\{PR_1,PR_2,...,PR_p,...PR_k\}$, $PR_p=\{rc_1.f_1, rc_2.f_2,...,rc_m.f_m\}$, the sum of the frequencies of all nodes for popular route is defined as:

$$SUMPR_p = \sum_{i=1}^{m} PR_p.rc_i.f_i, \quad (11)$$

Since the sum of the node frequencies indicates the route popularity, the popularity weight of the route is defined as:

$$w_{PR_p} = \frac{SUMPR_p}{\sum_{j=1}^{k} SUMPR_j}, \quad (12)$$

where *k* represents the total number of popular routes, PR_j represents one of the popular routes, and $\sum_{j=1}^{k} SUMPR_j$ represents the sum of the popular routes frequencies.

The anomaly judgment is performed by calculating the anomaly score of the trajectory and its corresponding *k* routes, which is defined as:

$$Score = \sum_{p=1}^{k} w_{PR_p} * GDDis(MT_i, PR_p), \quad (13)$$

where MT_i represents the testing trajectory, and PR_p represents the *p*-th popular route.

Pseudo-code for the main steps of trajectory anomaly detection is given in Algorithm 2. First, the weights of the popular routes are calculated according to the node frequency (line 2). Then, the corresponding popular routes of testing trajectory are traversed to calculate the anomaly score between the trajectory and the popular route (lines 3-12). Finally, trajectory anomaly judgment is performed by the sum of the anomaly scores between the trajectory and the *PRL*, where trajectories larger than θ are anomalous (lines 13-17). Meanwhile, from Algorithm 2, it can be seen that the time complexity of the PRTP algorithm mainly depends on two aspects: the loop of trajectory data and the loop of popular routes. Therefore, the time complexity of the PRTP algorithm is $O(n \cdot k)$, where n represents the number of *SD* trajectory datasets, and *k* represents the total number of popular routes. In most cases, the value of k is usually small, so the time complexity can be approximated to O(*n*).

Algorithm 2 Anomalous trajectory detection method using popular routes

Input: *gridLabel*, top-*k* Routes *PR*, Score threshold θ.

Output: Traanomalous(A dataset of anomalous trajectories).

```
1.  Score ← 0
2.  WeightList ←Calculate the frequency weights of each popular route according to equation (12)
3.  for i in gridLabel do
4.     PRL ← Get the grid sequences of the popular routes
5.     for j in PRL do
6.        LocMatrix_{MT_i} ← get the trajectory access matrix
7.        LocMatrix_{PR_j} ← get the route traversal matrix
8.        Calculate the LocMatrix_{common} according to equation (9)
9.        calculate GDDis according to equation (10)
10.       Score_j ← GDDis ×WeightList[j]
11.       Score ←Score +Score_j
12.    end for
13.       if Score > θ than
14.          put trajectory i into Traanomalous
15.       end if
16. end for
17. return Traanomalous
```

V. Experiments

In this section, an empirical evaluation of the proposed method is provided. All experiments are implemented with Python3 and conducted on a computer equipped with an Intel(R) Core(TM) i5-8250U CPU @ 1.80 GHz and 8GB main memory and running Windows 10 operating system.

A. Dataset

This study used the real taxi dataset provided by Piorkowski et al. [37], which contains all spatial locations of more than five hundred taxis in the San Francisco Bay Area in a month. The GPS trajectory records the latitude and longitude, timestamp, and corresponding passenger occupancy status of each taxi location point. In this study, the fixed starting and ending places selected were the airport and the central residential area. To reduce the noise and redundancy in the original trajectory data, it was assumed in the data preprocessing that only the trajectories of occupied taxis are valid. According to the above requirements, three pairs of SD trajectories (T-1 to T-3) were selected from the 463,860 trajectories extracted from 11.22 million GPS points, and whether each trajectory was anomalous or not was manually marked. The marked dataset is used to evaluate the detection accuracy of the PRTP algorithm.

B. Parameter Setting

Trajectory detection is essentially a binary classification problem, and *Recall* and *F-Score* are two important metrics to evaluate its performance. Therefore, these two metrics are used in this paper to judge the effectiveness of PRTP anomaly detection, which are defined as:

$$Precision = \frac{TP}{TP+FP}, \tag{14}$$

$$Recall = \frac{TP}{TP+FN}, \tag{15}$$

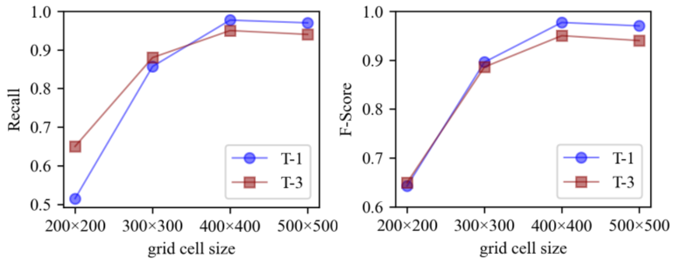
$$F-Score = \frac{2\times Precision\times Recall}{Precision+Recall}, \tag{16}$$

where *TP* (True Positive) denotes the number of detected anomalous trajectories, *TN* (True Negative) denotes the number of detected normal trajectories, *FP* (False Positive) denotes the number of normal trajectories incorrectly detected as anomalous, and *FN* (False Negative) denotes the number of anomalous incorrectly detected as normal [13].

To better investigate the effect of parameter settings in the PRTP algorithm on the detection result, this study conducted parameter setting experiments on the two largest datasets T-1 and T-3. There are three parameters in the detection stage of the PRTP, the grid size $\mathcal{M}$, the number of popular routes *k*, and the Score threshold θ. Therefore, this study experimentally investigates the effects of these three parameters on the *Recall* and *F-Score* of detection to determine the optimal values of the parameters.

1) Varying the size of grid cells. The setting of grid cell size determines the cell size when meshing the road network. The smaller the grid cell size, the higher the number of grids in the mapping trajectory. This makes the small differences to be magnified, which affects the detection accuracy. If the grid cell size is too large, there will be more trajectories with the same grid sequence in the dataset, which makes it more difficult to compare the similarity between the trajectory and the popular routes. Therefore, the reasonable setting of grid cell size is crucial to the performance of PRTP algorithm. Referring to previous studies [23,24], the performance variation is considered when the size of grid cells increases from 200m × 200m to 500m × 500m in this experiment. Note that the grid cell size set in this paper refers to the length and width of each small square.

In Fig. 7, the changes in *F-Score* and *Recall* for different grid sizes are illustrated. It can be seen that on the two different datasets, the maximum *F-Score* is obtained when the grid cell size is 400×400m, and the *Recall* also reaches the maximum at this time.



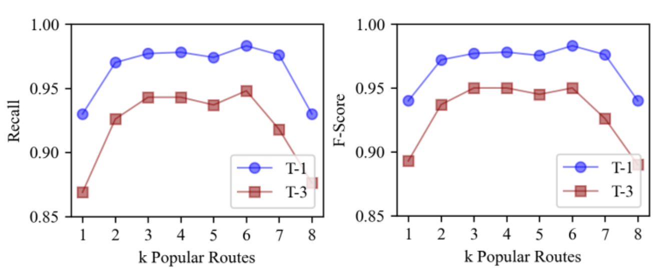
(a) (b)

Fig. 7. The Recall and F-Score of grid cells of different sizes.

2) Varying the number of popular routes. The number of popular routes *k* is an important parameter in the PRTP algorithm since the value of *k* determines how many routes can be used to calculate the anomaly score of the testing trajectory. Within a certain number of ranges, the detection results may be more accurate when more popular routes are taken for comparison. However, the testing trajectory needs to be compared with the routes one by one during anomaly judgment, so the detection time will increase linearly with *k*. Referring to the previous study [32], the range of *k* is set between 1 and 8.

Combined with the above definitions and calculation equations, it can be known that in the PRTP method, the node frequency is obtained from the node frequency graph of each popular route in advance, and when calculating the *Score* of the trajectory, each popular route is given the corresponding popularity weight and the popular route is also selected according to the order of node frequency. Fig. 8 shows the changes of *F-Score* and *Recall* for different *k* values. The results show that the *Recall* and *F-Score* of anomaly detection in the range of 1 to 6 generally have an upward trend as the *k* increases, when *k*>6, the detection performance gradually decreases as the *k* increases, and it can be seen that when *k*=6, the *F-Score* on the datasets T-1 and T-3 both reach the maximum, and the *Recall* also reaches the peak within the detection range.

3) Varying the Score threshold. In the last step of the PRTP algorithm, the parameter θ directly determines whether a trajectory is anomalous or not, if the *Score* of the trajectory is greater than θ, it is defined as anomalous, otherwise the trajectory is defined as normal. Therefore, it is necessary to study the influence of the value of θ on the performance of the algorithm. Fig. 9 shows the distribution of the *Score* values of the trajectories on the two datasets according to equation (13).



(a) (b)

Fig. 8. The Recall and F-Score of different popular routes.

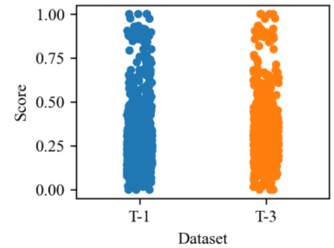
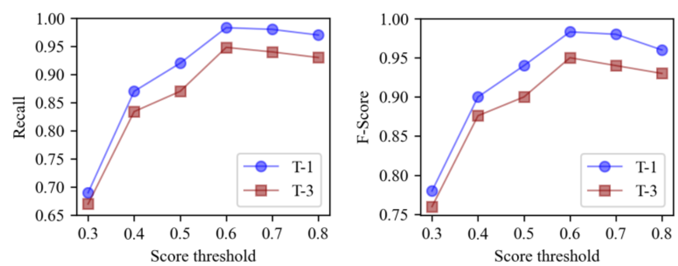


Fig. 9. The Score calculated by the PRTP algorithm.

From Fig. 9, it can be seen that most of the *Score* values on datasets T-1 and T-3 are less than 0.6, so based on the density distribution of *Score* values in the range of 0 to 0.6, 0.3 can be set as the lower limit of the range of θ. Moreover, the *Score* values of a small number of trajectories are distributed in the range of 0.6 to 1.0, and these points are likely to be the scores of anomalous trajectories, so 0.8 can be selected as the upper limit of the range of θ. Therefore, the article focuses on the effect of θ in the range of 0.3 to 0.8 on performance.

It can be seen from Fig. 10 that when the range between 0.3 and 0.6, as the threshold θ increases, the *Recall* of anomaly detection on datasets T-1 and T-3 gradually increases. Then, the *Recall* gradually decreases with the increase of θ, and when θ=0.6, the Recall achieves the maximum within the detection range of θ. Meanwhile, the *F-Score* shows the same variation trend, and the maximum value is obtained when θ=0.6. Therefore, considering the variation trend of the *Recall* and *F-Score*, θ=0.6 is chosen as the optimal value of the anomaly threshold.



(a) (b)

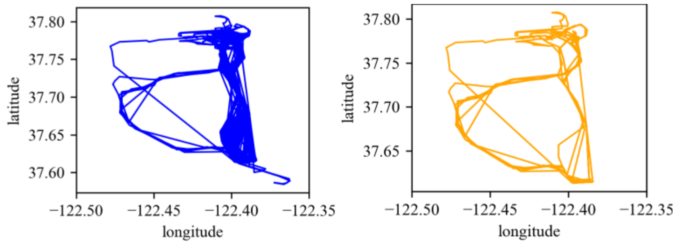
Fig. 10. The Recall and F-Score of different Score thresholds.

Finally, through the above analysis of the three variable parameters and the variation trends of *Recall* and *F-Score*, this paper determines the optimal values of these parameters as $\mathcal{M}$=400×400, *k*=6, and θ=0.6.

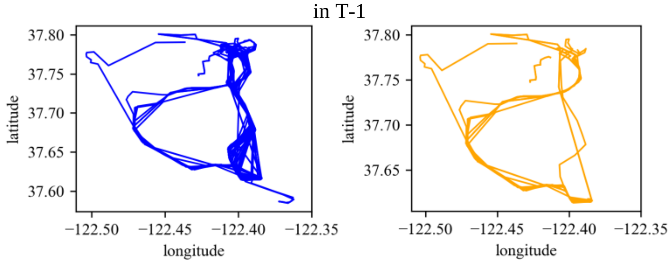
C. Visual Display of Anomalous Trajectory Detection

Anomaly detection is performed under the optimal values of the parameters. To understand the visualization result more clearly, this subsection selects some trajectories on the two largest datasets T-1 and T-3 for experiments to investigate the accuracy of PRTP. Fig. 11 shows the visualization result of anomaly detection on the real taxi trajectory datasets, where the solid blue lines represent all trajectories in the selected dataset and the solid orange lines show the anomalous trajectories detected by the PRTP algorithm.

It can be seen from Fig. 11(b) and Fig. 11(d) that the PRTP algorithm can fully detect obvious detour anomalous trajectories. Since the PRTP algorithm comprehensively considers the moment of the trajectory point, the time span of the trajectory, and the different frequency values in the detection process, it can correctly detect the local detour anomalous trajectories and global detour anomalous trajectories at different time periods. Therefore, these visualization results confirm the effectiveness of the PRTP algorithm.



(a) The selected trajectories in T-1 (b) The detected anomalous trajectories in T-1



(c) The selected trajectories in T-3 (d) The detected anomalous trajectories in T-3

Fig. 11. The result of anomalous trajectory detection on T-1 and T-3.

D. Comparative Evaluation

In the comparative experiments in this paper, to further verify the superiority of the PRTP, the study takes the ATDC [23], TPRO [32], and iBAT [24] as baselines to compare with PRTP. The basic idea of ATDC is to process trajectories by the grid. Then, it uses a distance calculation method to study different anomalous trajectory patterns and adopts an anomalous trajectory detection and classification method for real trajectory data. The iBAT algorithm employs an isolation mechanism to find anomalies in trajectories. Meanwhile, the basic idea of TPRO is to calculate the similarity between trajectories using edit distance in the spatiotemporal dimension. Therefore, the trajectory processing methods and research problems of these three algorithms are similar to this study.

Here, this module focuses on evaluating and comparing the anomaly detection effectiveness of the four algorithms by *F-Score* and *Recall*. From Fig. 12, it can be seen that among the baseline algorithms, ATDC achieves a higher *F-Score* in T-2, which indicates it has a good effect on the detection of global spatiotemporal anomalies. In T-1 and T-3 with more peak trajectories, TPRO shows good detection results due to its spatiotemporal sensitivity. And compared with the baseline algorithms, the PRTP algorithm obtains the highest *F-Score* on all datasets, and has more stable detection results.

Similarly, from Fig. 13, the *Recall* values of the four algorithms show the same trend, PRTP obtains superior detection results on all datasets. It can be seen that the *Recall* of the PRTP algorithm is the most ideal, while the *Recall* of the iBAT algorithm is the lowest, indicating that the iBAT algorithm more incorrectly detects anomalies as normal occurrences on the datasets. Meanwhile, the experiment results show that the PRTP algorithm can play a good role in detecting global anomalies as well as complex local anomalies. Therefore, this group of comparison experiments shows that PRTP can better detect anomalous trajectories compared to the baseline algorithms.

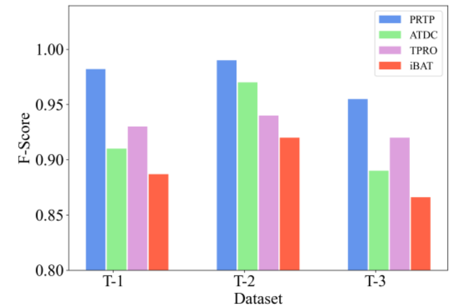


Fig. 12. The F-Score of PRTP, ATDC, TPRO, and iBAT on datasets.

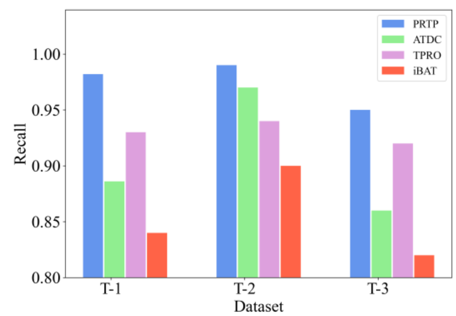


Fig. 13. The Recall of PRTP, ATDC, TPRO, and iBAT on datasets.

In addition to comparing the detection effectiveness of the four algorithms, this paper further compares the time cost of these algorithms. Among them, each algorithm runs ten times in the same environment and the average of the running times is taken as the final result. From Fig. 14, it can be seen that the PRTP algorithm also has an absolute advantage in time consumption. The time loss of ATDC is also ideal, but for the TPRO algorithm, since it chooses a small time interval when dividing the time period, it consumes more time during the whole algorithm detection.

Therefore, with optimal values of the parameters, PRTP achieves high values of both *F-Score* and *Recall* for anomaly detection results on the datasets. Meanwhile, the experimental results also show that the PRTP algorithm has a significant advantage in time loss.

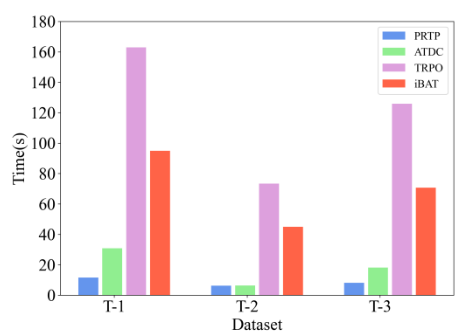


Fig. 14. The running time of PRTP, ATDC, iBAT, and TPRO on datasets.

VI. Discussion

In different application scenarios, most of the existing anomalous trajectory detection methods do not adequately integrate the location, time, and structure of the trajectories, and also ignore the influence of the time period on the route. Traditional algorithms usually compare the similarity of trajectories only in spatial or spatiotemporal dimensions, which is not applicable to anomaly detection of spatial trajectories with complex distribution. Compared with previous studies, the PRTP algorithm fully considers the location, time period, and structure of the testing trajectory in anomaly detection. The trajectory dataset is divided into different groups according to the division of time periods. Through the distribution of trajectories in different groups, the popularity of the route in different time periods is accurately obtained. Among them, to exclude the interference of trajectories with consistent running trend but long time-consuming to the route popularity, this study proposes the calculation method of spatiotemporal frequency, which avoids defining the congested roadway as a popular route. In the anomaly detection stage, to not be limited to the distribution location of the grid cells, this paper combines the location matrix and structure matrix of the trajectory to make anomaly judgments, which effectively detect anomalous spatiotemporal trajectories and loop travel trajectories, and also accurately identify normal trajectories bypassing congested roads. Experiments on real trajectory datasets show that the method can effectively detect trajectory anomalies with higher accuracy. The method can be applied in urban traffic road condition detection and traffic management, providing a new detection scheme for trajectory data mining.

VII. Conclusion

To improve detection accuracy and efficiency, this paper proposes an anomalous trajectory detection method using popular routes in different traffic periods. First, the method grids the trajectories and uses mapped trajectories for the study. Second, different time periods are divided according to the distribution of traffic flow, and the spatiotemporal node frequency values are obtained by combining the trajectory attributes, to obtain the popular routes dynamically. Finally, the distance formula is proposed for trajectory anomaly detection by combining trajectory location and trajectory structure. The proposed method is validated on real taxi GPS data, and it shows remarkable performance in experiments. Also, the method shows its potential in innovative applications such as taxi driving fraud detection.

However, the proposed method has a few limitations. In detecting anomalous trajectories, only the spatiotemporal properties of the trajectories are considered, and the exact location where the trajectory anomaly occurs and the time when the anomaly ends cannot be determined online. Therefore, in the follow-up study, this aspect will be broadened. For example, it is possible to detect anomalous sub-trajectories by combining time windows or decay factors while considering spatiotemporal attributes, to accurately locate where the anomalies occur and classify the anomalies of the trajectories based on the result.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant Nos. 62272006, 61972439).

References

[1] A. Belhadi, Y. Djenouri, G. Srivastava, D. Djenouri, A. Cano, and J. Lin, "A two-phase anomaly detection model for secure intelligent transportation ride-hailing trajectories," IEEE Transactions on Intelligent Transportation Systems, vol. 22, pp. 4496-4506, 2020.

[2] Q. Yu, Y. Luo, C. Chen, and S. Chen, "Trajectory similarity clustering based on multi-feature distance measurement," Applied Intelligence, vol. 49, pp. 2315-2338, 2019.

[3] Y. Ding, W. Zhang, X. Zhou, Q. Liao, Q. Luo, and L. Ni, "FraudTrip: Taxi fraudulent trip detection from corresponding trajectories," IEEE Internet of Things Journal, vol. 8, pp. 12505-12517, 2020.

[4] A. Giannoula, A. Gutierrez-Sacristán, Á. Bravo, F. Sanz, and L. Furlong, "Identifying temporal patterns in patient disease trajectories using dynamic time warping: a population-based study," Scientific Reports, vol. 8, pp. 4216, 2018.

[5] J. Hao, H. Zhang. "Ship trajectory anomaly detection based on TCN model", In International Conference on Guidance, Navigation and Control. Singapore: Springer Nature Singapore, pp. 3515-3525, 2022.

[6] J. Mao, T. Wang, C. Jin, and A. Zhou, "Feature grouping-based outlier detection upon streaming trajectories," IEEE Transactions on Knowledge and Data Engineering, vol. 29, pp. 2696-2709, 2017.

[7] Y. Shi, D. Wang, Z. Ni, H. Liu, B. Liu, and M. Deng, "A sequential pattern mining based approach to adaptively detect anomalous paths in floating vehicle trajectories," IEEE Transactions on Intelligent Transportation Systems, vol. 23, pp. 18186-18199, 2022.

[8] Y. Wang, K. Qin, Y. Chen, and P. Zhao, "Detecting anomalous trajectories and behavior patterns using hierarchical clustering from taxi GPS data," ISPRS International Journal of Geo-Information, vol. 7, p. 25, 2018.

[9] W. Zhou, Y. Yu, Y. Zhan, and C. Wang, "A vision-based abnormal trajectory detection framework for online traffic incident alert on freeways," Neural Computing and Applications, vol. 34, pp. 14945-14958, 2022.

[10] Z. Hong, Y. Chen, and H. Mahmassani, "Recognizing network trip patterns using a spatio-temporal vehicle trajectory clustering algorithm," IEEE Transactions on Intelligent Transportation Systems, vol. 19, pp. 2548-2557, 2017.

[11] X. Cai, B. Aydin, A. Ji, and R. Angryk, "A framework for local outlier detection from spatio-temporal trajectory datasets," International Conference on Pattern Recognition, pp. 5682-5689, 2021.

[12] F. Meng, G. Yuan, S. Lv, Z. Wang, and S. Xia, "An overview on trajectory outlier detection," Artificial Intelligence Review, vol. 52, pp. 2437-2456, 2019.

[13] S. Qian, B .Cheng, J.Cao, G.Xue, Y. Zhu, J. Yu, and T.Zhang, "Detecting taxi trajectory anomaly based on spatio-temporal relations," IEEE Transactions on Intelligent Transportation Systems, vol. 23, pp. 6883-6894, 2021.

[14] G. A. Gomes, E. Santos, C. A. Vidal, T. L. C. da Silva, and J. Macedo, "Real-time discovery of hot routes on trajectory data streams using interactive visualization based on gpu," Computers & Graphics, vol. 76, pp. 129-141, 2018.

[15] A. Belhadi, Y. Djenouri, G. Srivastava, A. Cano and J. Lin, "Hybrid group anomaly detection for sequence data: application to trajectory data analytics," IEEE Transactions on Intelligent Transportation Systems, vol. 23, pp. 9346-9357, 2021.

[16] F. Luan, Y. Zhang, K. Cao, and Q. Li, "Based local density trajectory outlier detection with partition-and-detect framework," International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, pp. 1708-1714, 2017.

[17] H. Wu, W. Sun, and B. Zheng, "A fast trajectory outlier detection approach via driving behavior modeling," ACM on Conference on Information and Knowledge Management, pp. 837-846, 2017.

[18] H. Zhang, Y. Luo, Q. Yu, L. Sun, X. Li and Z. Sun, "A framework of abnormal behavior detection and classification based on big trajectory data for mobile networks," Security and Communication Networks, 2020.

[19] J.-G. Lee, J. Han, and X. Li, "Trajectory outlier detection: A partition-and-detect framework," IEEE 24th International Conference on Data Engineering, pp. 140-149, 2008.

[20] I. San Román, I. Martín de Diego, C. Conde, and E. Cabello, "Outlier trajectory detection through a context-aware distance," Pattern Analysis and Applications, vol. 22, pp. 831-839, 2019.

[21] S. Shuang, C. Yan, and Z. Jinsong, "Trajectory outlier detection algorithm for ship AIS data based on dynamic differential threshold," Journal of Physics Conference Series, vol. 1437, p. 012013, 2020.

[22] Q. Yu, Y. Luo, C. Chen and X. Wang, "Trajectory outlier detection approach based on common slices sub-sequence," Applied Intelligence, vol. 48, pp. 2661-2680, 2017.

[23] J. Wang, Y. Yuan, T. Ni, Y. Ma, M. Liu, G. Xu and W. Shen, "Anomalous trajectory detection and classification based on difference and intersection set distance," IEEE Transactions on Vehicular Technology, vol. 69, pp. 2487-2500, 2020.

[24] D. Zhang, N. Li, Z.-H. Zhou, C. Chen, L. Sun and S. Li, "iBAT: detecting anomalous taxi trajectories from GPS traces," in Proceedings of the 13th International Conference on Ubiquitous Computing, pp. 99-108, 2011.

[25] W. Yu, "Discovering frequent movement paths from taxi trajectory data using spatially embedded networks and association rules," IEEE Transactions on Intelligent Transportation Systems, vol. 20, pp. 855-866, 2018.

[26] H. Zhang, W. Huangfu and X. Hu, "Inferring the most popular route based on ant colony optimization with trajectory data," China Conference on Wireless Sensor Networks, pp. 307-318, 2017.

[27] J. Tang, Y. Song, H. J. Miller and X. Zhou, "Estimating the most likely space–time paths, dwell times and path uncertainties from vehicle trajectory data: A time geographic method," Transportation Research Part C: Emerging Technologies, vol. 66, pp. 176-194, 2016.

[28] X. Han, R. Cheng, C. Ma and T. Grubenmann. "DeepTEA: Effective and efficient online time-dependent trajectory outlier detection," Proceedings of the VLDB Endowment, vol. 15, pp. 1493-1505, 2022.

[29] D. Chen, Y. Du, S. Xu, Y. -E. Sun, H. Huang and G. Gao, "Online Anomalous Taxi Trajectory Detection Based on Multidimensional Criteria," 2021 International Joint Conference on Neural Networks, pp. 1-8, 2021.

[30] Y. Ge, H. Xiong, Z.-h. Zhou, H. Ozdemir, J. Yu and K. C. Lee, "Top-eye: Top-k evolving trajectory outlier detection," in Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 1733-1736, 2010.

[31] J. Shi, Z. Pan, J. Fang, P. Chao, "RUTOD: real-time urban traffic outlier detection on streaming trajectory," Neural Computing and Applications, vol. 35, pp. 3625-3637, 2021.

[32] J. Zhu, W. Jiang, A. Liu, G. Liu, and L. Zhao, "Time-dependent popular routes based trajectory outlier detection," in International Conference on Web Information Systems Engineering, pp. 16-30, 2015.

[33] J. Zhu, W. Jiang, A. Liu, G. Liu, and L. Zhao, "Effective and efficient trajectory outlier detection based on time-dependent popular route," World Wide Web, vol. 20, pp. 111-134, 2017.

[34] Yuqing Yang, J C, Haifeng Yang, Jifu Zhang, Xujun Zhao. "TAD: A trajectory clustering algorithm based on spatial-temporal density analysis," Expert Systems with Applications, vol. 139, p. 112846, 2020.

[35] L. He, X. Niu, T. Chen, K. Mei, and M. Li, "Spatio-temporal trajectory anomaly detection based on common sub-sequence," Applied Intelligence, vol. 52, pp. 7599-7621, 2021.

[36] E. O. Eldawy, A. Hendawi, M. Abdalla, and H. Mokhtar, "FraudMove: Fraud drivers discovery using real-time trajectory outlier detection," ISPRS International Journal of Geo-Information, vol. 10, p. 767, 2021.

[37] N.Michal Piorkowski, Matthias Grossglauser, CRAWDAD dataset epfl downloaded from https://crawdad.org/epfl/mobility/20090224J.