International Journal of Geo-Information

MDPI

Article

Privacy-Preserving Trajectory Data Publishing by Dynamic Anonymization with Bounded Distortion

Songyuan Li [1], Hui Tian [2], Hong Shen [1,*] and Yingpeng Sang [1]

[1] School of Computer Science, Sun Yat-sen University, GuangZhou 510275, China; lisy36@mail2.sysu.edu.cn (S.L.); sangyp@mail.sysu.edu.cn (Y.S.)
[2] School of Information and Communication Technology, Griffith University, Nathan 4111, Australia; hui.tian@griffith.edu.au
* Correspondence: shenh3@mail.sysu.edu.cn

Abstract: Publication of trajectory data that contain rich information of vehicles in the dimensions of time and space (location) enables online monitoring and supervision of vehicles in motion and offline traffic analysis for various management tasks. However, it also provides security holes for privacy breaches as exposing individual's privacy information to public may results in attacks threatening individual's safety. Therefore, increased attention has been made recently on the privacy protection of trajectory data publishing. However, existing methods, such as generalization via anonymization and suppression via randomization, achieve protection by modifying the original trajectory to form a publishable trajectory, which results in significant data distortion and hence a low data utility. In this work, we propose a trajectory privacy-preserving method called dynamic anonymization with bounded distortion. In our method, individual trajectories in the original trajectory set are mixed in a localized manner to form synthetic trajectory data set with a bounded distortion for publishing, which can protect the privacy of location information associated with individuals in the trajectory data set and ensure a guaranteed utility of the published data both individually and collectively. Through experiments conducted on real trajectory data of Guangzhou City Taxi statistics, we evaluate the performance of our proposed method and compare it with the existing mainstream methods in terms of privacy preservation against attacks and trajectory data utilization. The results show that our proposed method achieves better performance on data utilization than the existing methods using globally static anonymization, without trading off the data security against attacks.

Keywords: trajectory data; data publishing; privacy-preserving; bounded distortion; attack preventing

Citation: Li, S.; Tian, H.; Shen, H.; Sang, Y. Privacy-Preserving Trajectory Data Publishing by Dynamic Anonymization with Bounded Distortion. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 78. https://doi.org/10.3390/ijgi10020078

Academic Editor: Wolfgang Kainz and Bart Kuijpers
Received: 27 December 2020
Accepted: 13 February 2021
Published: 16 February 2021

1. Introduction

The development of information technology and mobile Internet, as well as people's demand for convenient life, has spawned a large number of location-based service (LBS) applications, enabling the majority of mobile client users to enjoy the high-quality positioning and recommendation services, while also producing a large number of trajectory data of mobile objects including vehicles.

Trajectory data contain abundant information of mobile objects in the time and space dimensions. When combining with relevant knowledge to these data for aggressive inference and analysis, massive privacy information may easily be extracted, which even poses a threat to personal safety [1].

Therefore, in order to prevent the leakage of trajectory privacy, trajectory privacy preservation emerges as an important topic that has attracted increasing attention recently. Trajectory privacy preservation technique is based on social relations and location information. It uses the user's mobile and scene relations to deal with the location association relation in the trajectory, so as to form the trajectory information of the specific preserving

target. Its main purpose is to both ensure high-quality services of trajectory data sharing and protect individual's trajectory privacy [2].

Recently, many kinds of trajectory privacy-preserving technologies have been proposed. Most of the existing technologies need to create false location information, which can protect trajectory privacy to a certain extent. However, the large number of false trajectories results in the trajectory information distorted seriously, making guarantee of the quality of service in data utility hard to achieve. It also makes statistical analysis on the modified data biased and possible service delay due to loss of network traffic information [3]. In order to overcome these problems, in this paper we propose a novel method of dynamic anonymization based on bounded distortion trajectories mixing. This method does not need to generate a large number of false locations and trajectories to achieve a static global anonymity as in the existing methods. Rather, it performs dynamic anonymization based on mixing real trajectory segments with least added noise hop-by-hop progressively in a localized manner to form synthetic trajectories for publishing.

Our method of trajectory mixing is to set temporal and spatial windows and select suitable real trajectories in each window to permute and generate synthetic trajectory within a desired bound of orientation divergence. For example, for three trajectories A, B and C in four windows $A = A_1$—A_2—A_3—A_4, $B = B_1$—B_2—B_3—B_4, and $C = C_1$—C_2—C_3—C_4, our trajectory mixing produces three synthetic trajectories: $A' = B_1$—C_2—C_3—B_4, $B' = A_1$—A_2—C_3—A_4, and $C' = B_1$—A_2—A_3—A_4, where in each window, the orientation divergence between each real trajectory and its synthetic is within a given bound θ. Because each synthetic trajectory is composed of segments of real individual trajectories, it does not correspond to any individual real trajectory, therefore all real trajectories are effectively protected. At the same time, since each synthetic trajectory contains a sequence of segments of real trajectories within a orientation divergence, the published data will have similar statistical properties as the original and hence possess a good utility.

The main contributions of the paper are:

(1) We propose a novel dynamic anonymization method based on localized trajectory mixing to address the problem of privacy-preserving trajectory data publishing.

(2) We propose a novel framework for trajectory representation that facilitates efficient identification of intersections among trajectories, and an algorithm for computing trajectory intersection and individual mixing.

(3) Our method improves the data utility while preserving data privacy over the existing methods using the globally static anonymization technique.

(4) We conduct extensive experiments on real urban vehicle trajectory data to demonstrate the effectiveness of our method for privacy-preserving trajectory data publishing, and show that our proposed method achieves better data utility than the existing mainstream methods, without trading off the data security against attacks.

This paper is organized as follows. Section 2 discusses related work. Section 3 defines the basic concepts and attack model. Section 4 gives a comprehensive introduction to the idea, framework, algorithm and applications of our proposed method. Section 5 proposes an evaluation model of trajectory privacy methods, including the evaluation index of privacy protection and data utility. Section 6 presents experimental results based on the evaluation model, and comparison of our proposed methods with other protection methods. Section 7 concludes the paper with comments and future work.

2. Related Work

Many existing techniques are based on independent and identically distributed (i.i.d.) position sampling, sampling positions from random walks on grids, road networks or between points of interest, but the specific algorithms and operations are different.

Shokri et al. [4,5] proposed a uniform independent identical distribution method, which generates each false location independently from the uniform probability distribution and makes it have an identical distribution. Therefore, the false trajectory is a series of unrelated false positions.

The methods proposed by Chow et al. and Krumm et al. [6,7] can be summarized as follows: giving the probability distribution p of crowd movement, randomly walk on a series of positions with the probability distribution p, and finally generate a false trajectory with the selected positions.

Kato et al. [8] proposed a method to predict the random walk on the user's mobile trajectory, and then to predict the probability distribution $p(u)$ of the user's subsequent mobile trajectory. The probability distribution $p(u)$ was used to walk randomly on a series of positions. Finally, a false trajectory was generated from the selected positions.

The algorithms of these methods using a lot of false trajectories to cover up the real trajectory are similar, and the problems are also similar. The data fabricated by probability can easily be inferred to be invalid, and will cause the user's data to be totally useless in statistics.

Pingley et al. [9] presented a query-perturbation-based scheme that protects query privacy in continuous LBS even when user-identities are revealed, and Wang et al. [10] introduced a location privacy problem: Location-aware Location Privacy Protection (L2P2) problem to find the smallest cloaking area, these methods are too simple on privacy preservation. The main reason is that they use the trajectory of another user as the cover of data or even the trajectory of a non-user. This way of changing the trajectory attribution is very easy to be cracked by attackers and exposes the information of "innocent" persons directly. Although the methods are simple and effective in terms of reducing consumption and loss, they also have some shortcomings.

Most existing privacy-preserving trajectory data publishing technologies use generalization or disruption methods to deal with the published trajectory to conform to the k-anonymity model.

Machanavajjhala et al. [11,12] proposed an enhanced k-anonymity model, l-diversity model. The l-diversity principle requires that each k-anonymity group in a data table contain at least l different sensitive attribute values. The attacker infers that the probability of a recorded privacy message would be less than $1/l$.

Abul et al. [13,14] proposed (k, δ)-anonymity model based on the uncertainty of moving trajectory data. On the basis of the model, the problem of trajectory anonymity was treated by clustering. However, by analyzing the protection degree of (k, δ)-anonymity model, the model can only realize the k-anonymity of the trajectory just under the condition of $\delta = 0$.

Shin et al. [15,16] proposed an algorithm which divides a trajectory into a set of segments to ensure privacy. In these algorithm, the trajectory data are divided into several sections, which are anonymized to protect privacy of the trajectory and ensure the data utility above the desired level of quality of service.

The following are several more complex, mature and effective privacy-preserving methods, and all have their own advantages and disadvantages.

Darakhshan et al. [17] introduced the DP-WHERE method, which calls the Detailed Records (CDRs) database to generate different private composite databases, whose distribution is close to the real CDRs. However, CDRs are not equivalent to a complete location trajectory, because the location is known only when invoked.

Gursoy et al. [18] proposed a differentially private and utility preserving publication method for trajectory data. The method presents DP-Star, a methodical framework for publishing trajectory data with differential privacy guarantee as well as high utility preservation. From comparisons, the DP-Star significantly outperforms existing approaches in terms of trajectory utility and accuracy.

Zhao et al. [19,20] proposed a trajectory privacy-preserving method based on clustering using differential privacy. In these method, radius-constrained Laplacian noise is added to the trajectory location data in the cluster to avoid too much noise affecting the clustering effect, and they considered that the attacker can associate the user trajectory with other information to form secret reasoning attack, and proposed a secret reasoning attack model.

Proserpio et al. [21] introduced the wPINQ method, which achieves different privacy by calibrating the weights of some data records. They further proposed a method which generates synthetic data sets using Markov chain-Monte Carlo method, focusing on the graph of noise measurement given the number of triangles.

Bindschaedler et al. [3] proposed a metric for simultaneously capturing geographic and semantic features of real location trajectory. Based on these statistical metrics, a privacy-preserving generation model is designed to synthesize location trajectories. These trajectories may be the trajectories of some mobile individuals whose lifestyles are consistent and meaningful.

In recent years, many new trajectory privacy-preserving methods have been proposed.

Dai et al. [22] proposed a trajectory privacy-preserving method based on region partitioning, which mainly confuses attackers by sending pseudo-query points, and hides multiple query points in the same sub-region by using the region partitioning method covering the real trajectory of users, so that attackers can not reconstruct the real trajectory of users, thus protecting the privacy of users.

Sun et al. [23] proposed a privacy-preserving algorithm based on multi-characteristics of trajectory. The algorithm takes into account the uncertainty of trajectory data, and integrates the differences of direction, speed, time and space as the basis of trajectories in the same cluster set in the process of trajectory clustering and the utility of trajectory data after protection.

Zhang et al. [24] proposed a trajectory privacy-preserving method based on multi-anonymity. This method deploys n anonymity devices between users and location service providers. Each query is given a pseudonym. Combining with Shamir gate scheme, the user's query content is divided into n parts. The n parts of information are randomly distributed to n anonymity devices, and then sent to the provider after anonymity processing.

Zhu et al. [25] established a LBS trajectory privacy-preserving model for anonymous groups based on differential privacy. The model uses the idea of noise anonymous group to overcome the disadvantage of over-reliance on privacy budget of existing algorithms. At the same time, it ensures the quality of service of users through the idea of user trajectory partition and location.

Li et al. [26,27] proposed a privacy-preserving publishing method for trajectory data based on data partitioning. With the passage of time, the algorithm can effectively process the trajectories in each data partition without recalculating the published trajectories, thus effectively reducing the computational cost. It has efficient trajectory scanning, clustering and privacy-preserving functions.

From the perspective of technology implementation, the above trajectory privacy-preserving technology can be summarized into three kinds: trajectory anonymity-based technology, falsed trajectory-based technology and differential privacy-based technology. The main advantages, disadvantages and representative technologies of the three typical kinds are shown in Table 1.

Table 1. Three kinds of trajectory privacy-preserving technology

Types	Advantages	Disadvantages	Representative Technologies
Trajectory anonymity-based	Computation cost is medium, Implementation is simple	Resolution of trajectory data is distorted, Can be attacked by data feature inference	[24,27]
ine Falsed trajectory-based	Computation cost is small, Location information is accurate	Utility of trajectory data is reduced, Application depends on location model	[4,8]
ine Differential privacy-based	Comlete privacy protection, High service availability	Computation cost is high, Optimization methods need to be designed for applications	[18,19]

It should be noted that in the above methods, as the accumulation of false trajectories, trajectory offset, random location distribution added to the trajectory data, the difference of the trajectory data from its original grows continuously, making it hard for these methods to have a bounded data utility that is required by many real applications.

3. The Attack Model

3.1. Notations

We define the basic terminologies used in this paper below, and the mathematical symbols in Table 2.

Table 2. Important notations in this work.

Notations	Description
tr	trajectory
(x, y)	two spatial dimensions
p_k	position of tr at time t_k
$Dist(tr_1[t_k], tr_2[t_k])$	distance between $tr_1$and tr_2 at the time t_k
δ	threshold of trajectories similarity
ID	Identify of tr
$[t_{start}, t_{end}]$	time interval of trajectory
$tr[t_{start}, t_{end}]$	trajectory segment in $[t_{start}, t_{end}]$
$D(tr)$	trajectory database
$D_s(tr)$	the sampling database of $D(tr)$
$D_p(tr)$	the protected database of $D(tr)$ for publishing
h	a function transform $D_s(tr)$ to $D_p(tr)$
f	individual identity transformation function
g	location information transformation function

Definition 1. *Trajectory (tr). A trajectory is a path in the three-dimensional space (two spatial dimensions and one temporal dimension), represented by* $tr = \{p_1, p_2, \ldots, p_m\}$. *A point (position) of tr* $p_k = (x_k, y_k, t_k)$ *, where* x_k, y_k *are longitude and latitude,* t_k *is time,* $t_1 < t_2 < \cdots < t_m$, *and m is the number of sampling points.*

A trajectory is identified by a unique number called Identify (ID).

We use $D(tr)$ to denote the database of trajectories: $D(tr) = \{(QI, tr_i)\}$, $|D| = n$, $1 \leq i \leq n$. n is the number the trajectory individuals, $D_s(tr) \subseteq D(tr)$ is the trajectory sampling database, and $D_p(tr)$ is the protected $D(tr)$ for publishing.

3.2. Two-Level Transformation

Let $h : D_s(tr) \rightarrow D_p(tr)$ be a function that transforms the trajectory database $D_s(tr)$ to the trajectory publishing database $D_p(tr)$ to achieve privacy-preserving trajectory publishing.

h can be decomposed into two levels of of transformation: individual identity transformation f and location information transformation g, and $h = f \cdot g$.

For protecting the individual identity, we transform the identity information of the trajectory individuals, $f : u \rightarrow v$, f is the mapping function between individual u and v, which is not published. After individual identity protection, the 4-tuples of each trajectory point are transformed from (u, x, y, t) to (v, x, y, t).

For protecting the spatial and temporal information, (x, y) is transformed to (x', y') using $g : (x, y) \rightarrow (x', y')$.

The distance between trajectories is measured by Euclidean distance. Euclidean distance between two trajectories tr_1 and tr_2 at time t ([28,29]) is:

$$Dist(tr_1[t_k], tr_2[t_k]) = \sqrt{(tr_1[t_k].x - tr_2[t_k].x)^2 + (tr_1[t_k].y - tr_2[t_k].y)^2)} \quad (1)$$

In the time range of $[t_{start}, t_{end}]$, the distance of $tr_1$and tr_2 is:

$$Dist(tr_1, tr_2) = \sum_{t=t_{start}}^{t_{end}} Dist(tr_1[t], tr_2[t]) \quad (2)$$

Definition 2. *Intersecting of m trajectories. Within the time range $[t_{start}, t_{end}]$, m trajectories tr_1 to tr_m are intersecting if the distance between the m at any time t_k in $[t_{start}, t_{end}]$ is less than δ, i.e., $Dist(tr_1[t_k], tr_m[t_k]) \leq \delta$, where δ is the threshold of the Intersection of m Trajectories.*

3.3. Attack Model

According to the application scenario of trajectory data publishing, we analyze the attack model in trajectory data publishing.

By analyzing the $D_p(tr)$, the attacker can restore or partially restore the trajectory database $D_s(tr)$, this attack is called trajectory privacy attack. Trajectory privacy attack function $h^{'} = h^{-1} : D_p(tr) \rightarrow D_s(tr)$, which is the inverse process of trajectory protection. The goal is to restore $(v, x^{'}, y^{'}, t)$ to (u, x, y, t), where u is the true ID of tr, v is the falsified ID of tr.

Accordingly, the trajectory privacy attacks can be divided into two levels, location information inference and individual identity inference, $h^{'} = g^{'} \cdot f^{'}$.

$g^{'} : (x^{'}, y^{'}) \rightarrow (x, y)$. With the help of road network information, background information or other noise reduction methods, the attacker can infer the location information by $g^{'}$.

$f^{'} : (x, y, t) \rightarrow u$. On the basis of (v, x, y, t), the attacker can infer u from the information contained in (x, y, t), and establish the the association between the individual identity u and v.

There are different ways of implementing location inference function $g^{'}$ in different protection algorithms.

The implementation of individual identity inference function $f^{'}$ follows the general framework of inferring first the individual's address, and then the individual's identity through the address combining the background information, so as to establish the relationship between the individual's identity and the individual's trajectory.

Trajectory Inference (TrajInfer) is to determine the relationship between individual trajectory (QI, tr) and individual ID, where $tr = \{p_1, p_2, \ldots, p_m\}$. If trajectory inference is successful, trajectory privacy will be breached.

Position inference attack is to infer individual sensitive position information (such as home) based on trajectory data.

Attackers may find a user's trajectory information in the following ways [3,30,31] :

(1) Popular positions

The attacker tries to identify the most popular positions by computing the positions that have been visited most frequently between the home and work place. This attack performs the following ranking function:

$$Top(Count((x, y)))$$

where $Count(\cdot)$ calculates the number of position (x, y) in all trajectories.

(2) The mega clusters

The attacker tries to identify the two largest clusters containing data points of the trajectory tr. This attack performs the following function:

$$Top(Clustering((x, y, t)|(x, y, t) \in tr))$$

Clearly, taking the data analysis and mining scenario for users' living habits as an example, we usually assumed that most of the user's location points will be distributed at home and in the workplace, as home and workplace are the most popular positions for all the users, they are returned as centres of the top 2 largest clusters.

(3) Popular positions within a time interval

The attacker tries to identify the popular positions that most users visited during a particular time interval (t_{start}, t_{end}). This attacker performs the following calculation:

$$Top(Clustering((x, y, t)|t_{start} \leq t_k \leq t_{end}))$$

4. The Proposed Algorithm

To achieve the two-level transformation with location information tansformation and individual identity transformation for privacy protection, we propose the algorithm Trajectory Privacy Preservation by Dynamic Anonymization with Bounded Distortion (TPP-DABD) below.

4.1. Algorithm Outline

Our TPP-DABD algorithm contains the following four key steps:

1. Compute a time window and partition the region for anonymization.
2. Form trajectory pairs (denoted by P) with intersecting angle no greater than θ.
3. Introduce fewest dummy segments to pair the remaining trajectory segments within orientation divergence degree θ, resulting $\tilde{P}$ pairs.
4. Swap segments in each pair of $P \cup \tilde{P}$.

The algorithm can execute both online and offline as needed. For online execution, it repeats the above four steps as all trajectories progress, where the time window in each execution is computed dynamically according to the application needs and orientation of the trajectories. For offline execution, because the complete information of all trajectories are known, it will have all time windows computed in Step 1 and then do Steps 2~3 for each time window.

For Steps 2 and 3, for the purpose of algorithm efficiency, we use a greedy approach to compute a satisfactory number rather than an optimum number which is computationally too expensive to achieve.

4.2. Algorithm Description

To implement the above algorithm outline, our main idea is to segment the trajectory into windows and make the mixing of QI within a bounded distortion firstly based on the trajectory segments inside each window and then by introducing least dummy segments, so as to prevent the attacker from identifying the individual tracks..

The four steps of our algorithm are implemented by taking into account of the intersection patterns of trajectory segments as follows:

- Step 1: Compute a time window $[t_{start}, t_{end})$ and partition the region for anonymization. In order to determine the intersection quickly, we apply the method of gridding to approximately partition the region (network of trajectories) as follows. The principle of this step is to divide the whole region into two sets of grids, $grid1$ and $grid2$, where each set contains $d \times d$ squares, $grid1$ and $grid2$ are overlapped, the center of $grid1$ is a vertex of $grid2$. The distance between two points falling into the same grid is approximately considered to be d. Through this step, we grid the region, then we can use a simplifing way to judge whether the trajectory segments intersect by calculating whether their trajectory points are in the same grid.
- Step 2: Form trajectory pairs (denoted by P) with intersecting angle no greater than θ. In order to improve the utility of each trajectory data after the swapping, we firstly identify a maximum number of trajectory segment pairs whose intersecting angles are below the threshold θ, and then place the two segments in each pair respectively into $Left_P$ and $Right_P$. As shown in Figure 1, the principle of this step is to find the intersecting and divergent trajectory segments within the time window, and constrain the divergence degree of each pair of trajectory segments to a certain threshold θ. In this case, if the divergence degree between tr_a and tr_b is greater than θ, the entry (tr_a, tr_b) in the exchange matrix is set to infinity, so that the probability of exchange

between tr_a and tr_b is 0. Through this step, we can prevent the direction of a trajectory from large angle divergence, so as to improve the utility of trajectory data after mixing.

- Step 3: Introduce fewest dummy segments to pair the remaining trajectory segments within orientation divergence degree θ. For all the left alone (unpaired) segments, we apply k-means clustering to find the smallest k centroids that place all the segments into clusters of radius θ, and result $\tilde{P}$ pairs. The main purpose of this step is to make a pair of the remaining trajectory segments. The principle of this step is to introduce virtual segments, which are combined with residual trajectory segments to form a pair of trajectories with directional divergence (intersecting angle) within θ. As shown in Figure 2, in order to construct the virtual segment, the intersection point of the virtual segment and the trajectory segment should be determined first, and then the input and output point of the virtual segment should be constructed with the intersection as the center. In order to reduce the number of virtual segments as much as possible, adjacent virtual segments can be merged into one segment by smooth connection.
- Step 4: Swap each pair of segments in $P \cup \tilde{P}$. We take the intersection as the boundary according to a given probability to replace the front and back QI, and realize the trajectory segment swapping. The replacement also needs to consider the difference and balance. The trajectory segments are exchanged randomly according to a certain probability, which is realized through the exchange matrix. The principle of this step is to replace the ID of the trajectory segment after the trajectory intersection point according to a certain probability. Through this step, the exchange of trajectory identifier can be completed.

The detailed algorithm including the four steps is described in Algorithm 1.

Algorithm 1: Trajectory privacy preservation by dynamic anonymization with bounded distortion

Input: $D(tr) : [(u, x, y, t)]$
Output: return the $D^{'}(tr) : [(v, x^{'}, y^{'}, t^{'})]$
//establish randomization time window $[t_{start}, t_{end})$, the width of time window is a random number from random().
for $(t = t_{min}; t < t_{max}; t+ = random())$ {
$t_{start} = t; t_{end} = min(t + random(), t_{max})$;
//to form trajectory segments pairs(P) with orientation divergence degree not greater than θ.
P=Algorithm-2($D(tr)$,t_{start},t_{end}) ;
//introduce fewest dummy segments to pair the remaining segments within orientation divergence degree θ.
$\tilde{P}$=Algorithm-3($D(tr)$,t_{start},t_{end}, P);
//swap each pair of segments in $P \cup \tilde{P}$
$D^{'}(tr)$=Algorithm-4($D(tr)$,P,$\tilde{P}$) ;
}

Step 2 is implemented by Algorithm 2 (paring trajectory segments), which returns the trajectory segments pairs (P) with orientation divergence degree not greater than θ.

In Algorithm 2, the process of searching for intersecting trajectory segments with the orientation divergence degree not greater than θ and putting them into pair which will be exchanged by Algorithm 4, is shown in Figure 1.

Algorithm 2: Pairing trajectory segments

Input: $D(tr) : [(u, x, y, t)]$, t_{start}, t_{end}
Output: return the $pair(P)$ with orientation divergence degree not greater than θ
initialize $grid1$, $grid2$
  //Scan each trajectory point p
  for each point $p(x, y, t)$ in $D(tr)$
  if ($t >= t_{start}$ && $t < t_{end}$){
    $t_{start} = t; t_{end} = min(t + random(), t_{max})$;
    //establish randomization time window $[t_{start}, t_{end})$, the width of time window is random number random()
  for ($t = t_{min}; t < t_{max}; t+ = random()$) {
  for each p in $grid1$
    find its pairing point p' in $grid1$, where p,p' belong to the same segment and their intersecting angle is less than θ
    if no such p' can be found
      move p out of $g1$ and map it to $grid2$ based on ($p.x$,$p.y$).
  add dummy points to $grid2$ to pair every p in $grid2$.
   }
    //Pair trajectory points with necessary dummies such that their orientation divergence degree is not greater than θ.
  }

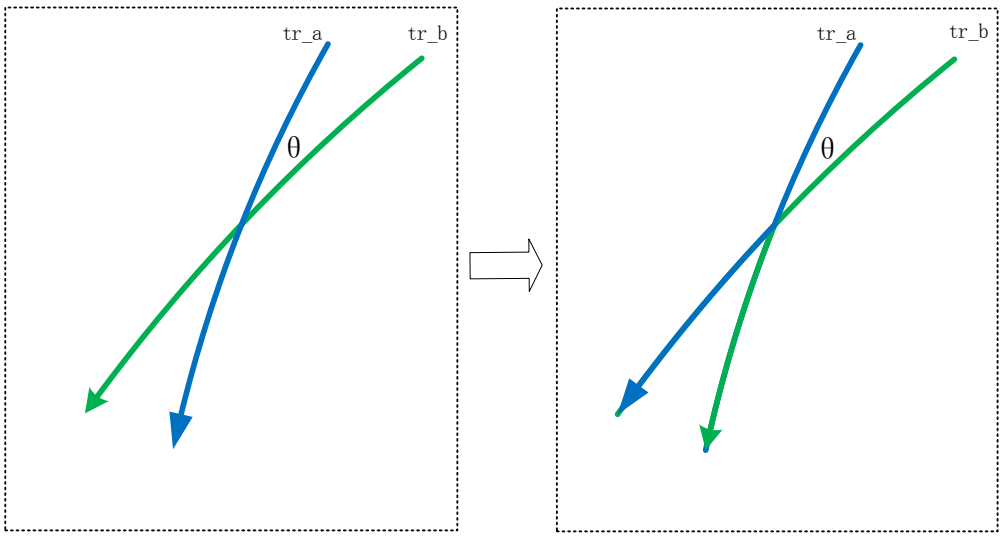

Figure 1. The process of exchanging intersecting trajectories.

As shown in Figure 1, the pairs of two intersecting trajectories are formed as follows:

(1) Select the time window $[t_{start}, t_{end})$, and initialize the grid of grid1 and grid2.

(2) In grid1 and grid2, the intersecting trajectory segments are calculated, and the angle between the intersecting trajectories is calculated at the same time.

(3) Through calculation, the trajectory segments whose intersecting angle does not exceed θ are found and form pairs, and the divergence degree of each pair of trajectory segments is limited within a certain threshold θ.

(4) In this case, if the divergence degree between tr_a and tr_b is greater than θ, the value of the exchange matrix (tr_a, tr_b) is set to infinity, so that the probability of exchange between tr_a and tr_b is 0.

Step 3 is implemented by Algorithm 3: pairing the remaining trajectory segments ($\tilde{P}$), which introduces fewest dummy segments to pair the remaining trajectory segments pairs ($\tilde{P}$) with orientation divergence degree not greater than θ.

Algorithm 3: Pairing the remaining trajectory segments ($\tilde{P}$)

```
Input: D(tr) : [(u, x, y, t)], t_start, t_end, pair(P)
Output: return the pair(P̃) with orientation divergence degree not greater than θ
 for each segment in D(tr)-P {
   compute the intersecting point for the dummy segment using the k-means
  algorithm
   set the random input and output points of the dummy segment while ensuring
  the intersecting angle is not greater than θ.
 }
 //using Greedy algorithm to construct the dummy segment.
 for each segment A in D(tr)-P
 if segment A is not in pair {
  construct the dummy segment by linking the input, intersecting and output
  points
   search for the remaining segment B which can intersect the dummy segment
  with an angle no greater than θ.
      if there exit such segment B, make B and the dummy segment a pair, and
  delete the existing dummy segment of segment B.
      otherwise, extend the dummy segment to the input point of dummy segment
  of B.
    }
 put all the dummy segments into D(tr).
 return D(tr);
 }
```

Algorithm 3 solves the problem of the remaining trajectory segments.

After the process of Algorithm 2, the remaining trajectory segments are divided into the following two types, one contains the intersecting segments (the orientation divergence is greater than θ), the other contains the non-intersecting segments. Both types can be processed by introducing dummy segments, so that the divergence of dummy segment and remaining segment is within θ.

These pairs will be exchanged by Algorithm 4, as shown in the Figure 2.

As shown in Figure 2, the process of constructing a virtual trajectory segment to form a trajectory segment with directional divergence no greater than θ is as follows:

(1) According to the remaining trajectory segments, the intersection point between each remaining trajectory segment and the virtual trajectory segment, the input and output point of the virtual segment are constructed.

(2) In grid1 and grid2, the intersecting trajectory segments are calculated, and the divergence of the intersecting trajectories is calculated at the same time.

(3) Through calculation, the trajectory segments whose divergence does not exceed θ are found and form pairs, and the divergence degree of each pair of trajectory segments is limited within a certain threshold θ.

(4) In this case, if the divergence degree between tr_a and tr_b is greater than θ, the value of the exchange matrix (tr_a, tr_b) is set to infinity, so that the probability of exchange between tr_a and tr_b is 0.

In the construction of virtual segment, the intersection of dummy segment and true segment should be determined first, and then the input point and output point of dummy segment should be constructed with the intersection as the center. In order to reduce the number of dummy segments as much as possible, adjacent dummy segments can be combined into one segment through smooth connection, as shown in Figure 3.

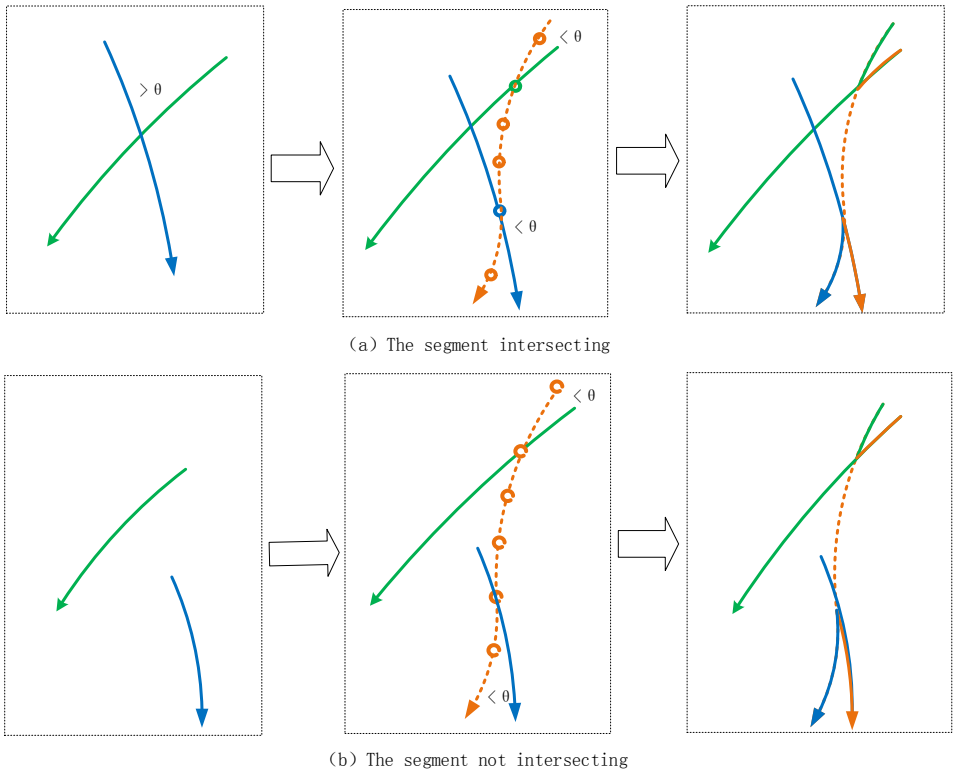

Figure 2. The process to pair the remaining trajectory segments.

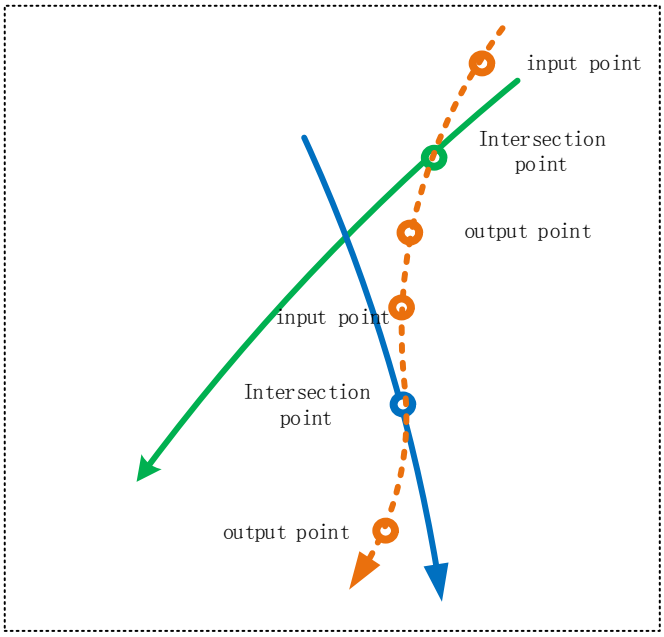

Figure 3. The adjacent dummy segments combined into one segment.

Step 4 is implemented by Algorithm 4: swapping each pair of segments in $P \cup \tilde{P}$.

Algorithm 4: Swapping each pair of segments in $P \cup \tilde{P}$

```
Input: D(tr) : [(u, x, y, t)], P, P̃
Output: return the D'(tr) : [(v, x', y', t')]
  //exchange the QI after the intersection according to a certain probability
  //the probability is determined by the exchange matrix M.
  for each (tra, trb) {
    select v randomly and exchange v to v'
    with the probability at the entry (tra, trb) in M, complete the following:
    for each point in tra and each point in trb {
      exchange the IDs between them
    }
    update M for next exchange.
  }
```

5. Metrics for Performance Evaluation

At present, in the research and practice of trajectory privacy-preserving data publishing technology for trajectory data statistical analysis application scenarios, there are two kinds of metrics are usually used to evaluate the performance of algorithms: plausible deniability and statistical dissimilarity.

The metric of plausible deniability is used to evaluate the degree of trajectory privacy preservation. In this paper, plausible deniability refers to the degree to which an attacker can infer at least one synthesis trajectory with the same credibility as the original real trajectory when he infers the trajectory after mixing.

The metric of statistical dissimilarity is used to evaluate the utility of trajectory data. In this paper, statistical dissimilarity refers to the statistical difference between the trajectory data set generated by the trajectory privacy-preserving method and the original trajectory data set.

The metrics used in this paper shown in Table 3.

Table 3. Metrics used in this paper.

Metrics	Meaning
IER	Inference Error Rate
$E(IER)$	Mathemaatical Expectation of IER
$SFRR$	Statistical Feature Retention Rate
$ERoTI$	Error Rate of Trajectory Inference
$E(ERoTI)$	Mathemaatical Expectation of $ERoTI$

5.1. Degree of Privacy Preservation

In this paper, the Inference Error Rate (IER) is used as a metric to evaluate the degree of trajectory privacy preservation. The IER refers to the ratio of the number of wrong trajectory inferences to the total number of inferences, which can be expressed as follows:

$$IER = \frac{The\ Number\ of\ Errors\ in\ Trajectory\ Inference}{The\ Number\ of\ Trajectory\ Inference} \tag{3}$$

The closer the IER is to 1, the better the trajectory privacy is protected.

According to the algorithm of TPP-DABD, the mathematical expectation of the IER depends on the number of exchanges (n) on the synthesized trajectory. Given k trajectories in the mixing group:

$$E(IER) = 1 - \frac{1}{k^n} \tag{4}$$

As shown in Figure 4, when the number of trajectory exchanges is 0, the original trajectory corresponds to the QI one by one. If the attacker can find the ID of the individual corresponding to the QI through the trajectory inference attack, the original trajectory information of the individual can be correctly identified, that is $E(IER) = 0$.

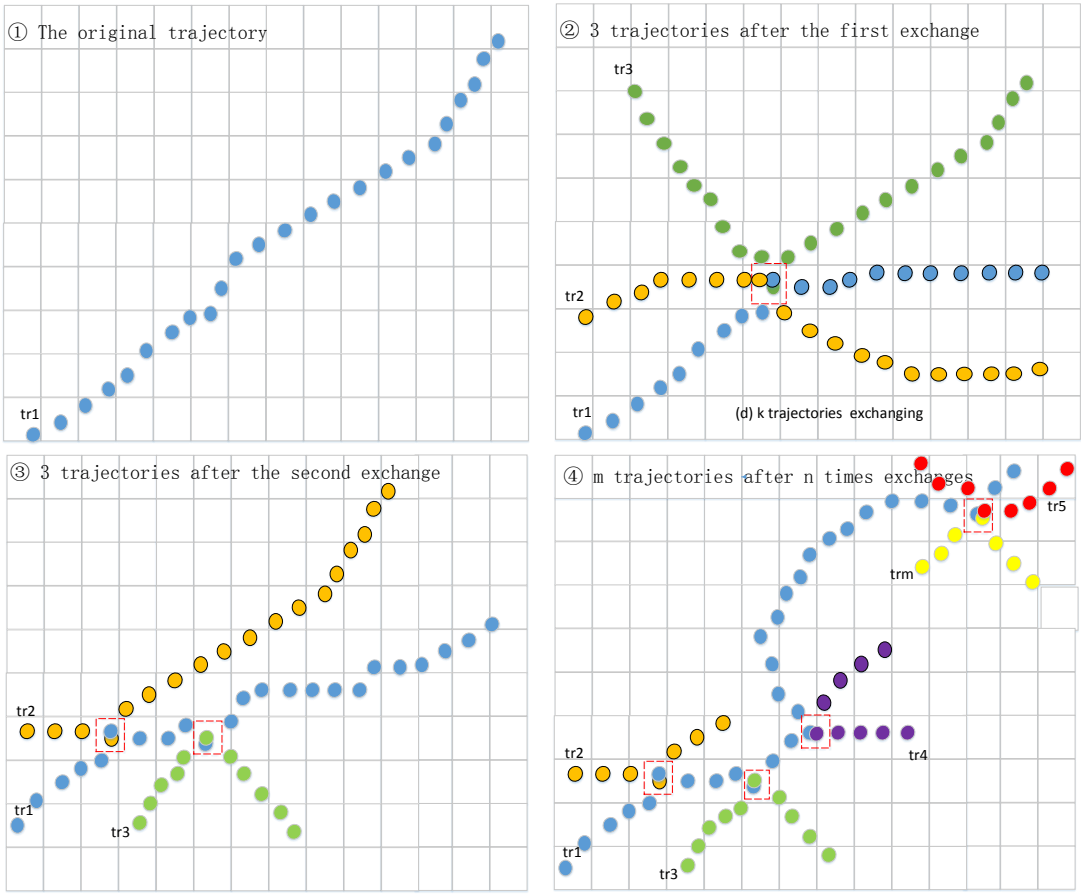

Figure 4. m trajectories after n times exchanges.

When the number of trajectory exchanges is 1, two trajectories are involved in the mixing on one intersection node. If the attacker can identify the ID through the trajectory inference attack, there is still $\frac{1}{2}$ probability that the original trajectory information of the individual can be correctly identified, that is $E(IER) = 1 - \frac{1}{2} = \frac{1}{2}$.

When the number of trajectory exchanges is n, there are k trajectories on n nodes involved in the mixing. If the attacker can find out the ID through the trajectory inference attack, there is still a $\frac{1}{k^n}$ probability that the original trajectory information of the individual can be correctly identified, that is $E(IER) = 1 - \frac{1}{k^n}$.

When the user's trajectory data is processed by the method of TPP-DABD, the mixed trajectory can no longer completely correspond to the trajectory information of each user in the original trajectory data.

However, the new trajectory data set based on TPP-DABD is consistent with the original trajectory data set in the location information distribution, and also conforms to the actual distribution of the location information of various users in the road network, which makes the trajectory data set formed after the synthesis still usable. Moreover, with the increase of the number of individual mixing at any time period, difficulty also increases for attackers to restore the synthesized trajectory to the original trajectory.

The mathematical expectations of IER with different n are listed in Table 4. When $n >= 7$, the expected IER will be more than 99%, which means a high degree of privacy preservation.

Table 4. The mathematical expectation of the IER.

ine n	$E(IER)$
ine 0	0.00
1	50.00%
2	75.00%
3	87.50%
4	93.75%
5	96.88%
6	98.44%
7	99.22%
8	99.61%
9	99.80%
10	99.90%
ine	

5.2. Utility of Trajectory Data

In some application scenarios, the evaluation method of data utility compares the difference between published data and original data. In this method, data utility and data privacy preservation are a pair of contradictions. A high utility of data in trajectory privacy preservation will limit the protection of trajectory privacy to a low level, especially in the long-term trajectory privacy-preserving data publishing.

In many application scenarios, the utility of trajectory data depends more on the statistical characteristics of real trajectory. The original statistical features can be retained after the trajectory data are protected. Unlike the method evaluating the difference between original and published data, a high retention of statistical features does not necessarily mean a low level of privacy protection, therefore it is a more reliable metric for data utility.

In this paper, we measure data utility based on statistical analysis, aiming to apply the data in scenarios like interest point extraction, location semantic annotation, map inference, business location finding, etc.

We use the Statistical Feature Retention Rate (*SFRR*) to evaluate the utility of trajectory data. The *SFRR* is defined as follows:

$$SFRR = \frac{FoST \cap FoOT}{FoOT} \tag{5}$$

in which $FoST$ are features of the synthetic trajectory, $FoOT$ are features of the original trajectory. The range of *SFRR* is $[0, 1]$. The closer it approaches 1, the higher the utility of trajectory data is.

Specifically, the following statistical metrics can be used [3,31] :

(1) The distribution of visits or the number of visitors per location.

(2) Distribution of visitors in the top 10 interest locations.

(3) The top n interest positions in the region.

(4) The user's time allocation.

In the utility evaluation, the same metric can be computed on the synthetic and original trajectory data respectively, and compared to analyze the retention of statistical features.

After the trajectory data are processed by the TPP-DABD algorithm, the coordinates and time of each position in the trajectory data set are not changed. Therefore, in terms of the statistical analysis on visiting distribution and user's time allocation, the features of synthetic and original trajectory are consistent, which means the SFRR is 1.

In Section 6, we will analyze and prove the above evaluation through experiments.

6. Experiments and Evaluations

In this section, we will use these metrics to compare our algorithm with some classical algorithms, through experimental results and analyze the advantages and disadvantages of these algorithms.

6.1. Datasets and Experimental Methods

The trajectory data set used in our experiments was based on the GPS data from taxis in Guangzhou City, China, including the trajectory data of about 2000 taxis for one day, and 340,000 trajectory location points. The total data size was 3 GB, and the trajectories were concentrated in the range of 112 to 114 degrees of east longitude and 22 to 23 degrees of north latitude.

The main process of the experiments was as follows: Firstly, the same original trajectory data set was selected, and different algorithms were run on the trajectory data set to form the published trajectory data set. Then, the validity of different algorithms in privacy protection was compared according to Inference Error Rate (IER), and the validity of different algorithms in data utility was compared according to Statistical Feature Retention Rate (SFRR).

The following classical trajectory privacy-preserving algorithms was compared with our algorithm:

(1) Uniform i.i.d. Sampling Method (UIIDSM) [4] . According to uniform probability distribution, each false position is generated independently and identically.

(2) Aggregated i.i.d. Sampling Method (AIIDSM) [4] . According to aggregated mobile Poisson distribution, each false location is generated independently and identically.

(3) Aggregated Random Moving Method (ARMM) [6,30]. False trajectories are generated by random moving on the set of positions after Poisson distribution.

(4) Random Movement Method of User Probability (RMMUP) [8]. On the basis of probability distribution $p(u)$ of user occurrence, a group of locations are formed, and random movement is carried out to generate false trajectories.

(5) Synthesize Trajectory Method based on Position Semantics (STMPS) [3]. False trajectories are generated based on position semantics.

(6) Trajectory data publishing based on data partitioning (DPCP) [27]. Trajectory data publishing under (k, δ) security constraints based on data partitioning.

6.2. Analysis of Privacy Preservation Degree

The goal of trajectory privacy preservation is to protect an individual's tracks, that is, to prevent his identity and relevant trajectory from being recognized.

To correctly infer an individual's track, the attacker needs to identify his trajectory as well as his identity. If the probabilities of these two types of identification are respectively π_1) and π_2), the mathematical expectation of the Error Rate of Trajectory Inference (ERoTI) is:

$$E(ERoTI) = 1 - \pi_1\pi_2 \tag{6}$$

(1) Probability of correctly identifying individual trajectories (π_1)

According to our individual mixing algorithm, suppose there are n times of mixing, the number of trajectories involved in the m-th mixing is k_m, and the attacker has found the individual identity for some trajectory, then the probability that the trajectory can be correctly identified by the attacker is:

$$\pi_1 = \frac{1}{k_1 k_2 \cdots k_n} \tag{7}$$

As shown in Figure 4, in the first window $n = 0$, the trajectory and the QI can be easily found, then the trace can be correctly identified, that is, $\pi_1 = 1$. In the second window $n = 1$, there are three trajectories involved in the mixing. If the individual identity is found by attacker, it is a probability of $\frac{1}{3}$ for the correct track to be identified, that is, $\pi_1 = \frac{1}{3}$. In the third window, $n = 2$, there are three trajectories involved in the first mixing, and four trajectories involved in the second mixing. At this time, after finding out the identity of the individual, it is a probability of $\frac{1}{12}$ for the track to be identified, that is, $\pi_1 = 8.3\%$.

When the number of trajectory exchanges is n, there are k trajectories involved in each time of mixing, then $\pi_1 = \frac{1}{k^n}$. As shown in Table 5, with the increase of k and n, π_1 converges to 0.

Table 5. The probability of correctly identifying individual trajectories (π_1).

ine π_1	$n = 2$	$n = 3$	$n = 4$	$n = 5$
ine $k = 1$	50.00%	33.33%	25.00%	20.00%
$k = 2$	25.00%	11.11%	6.25%	4.00%
$k = 3$	12.50%	3.70%	1.56%	0.80%
$k = 4$	6.25%	1.23%	0.39%	0.16%
$k = 5$	3.13%	0.41%	0.10%	0.03%
$k = 6$	1.56%	0.14%	0.02%	0.01%
$k = 7$	0.78%	0.05%	0.01%	0.001%
$k = 8$	0.39%	0.02%	0.002%	0.0003%
$k = 9$	0.20%	0.01%	0.0004%	0.0001%
ine				

(2) Probability of correctly identifying individual identity (π_2)

According to the attacking model, the probability of correctly identifying individual identity, π_2, mainly depends on the aggregation distribution of location points. The attacker can correctly identify the individual's identity in the original trajectory, but not necessarily in the synthesized trajectory.

Suppose the original trajectory $tr = \{p_1, p_2, ..., p_m\}$, and the synthesized trajectory is $tr^{'}$. If mixing starts from the point of m in $tr^{'}$, then $tr^{'} = \{\pi_1, \pi_2, ..., \pi_m, \pi^{'}_{m+1}, \pi^{'}_{m+2}, ..., \pi^{'}_{n}\}$. It is easy to see that:

$$Clustering(tr) \neq Clustering(tr^{'}), Count(tr) \neq Count(tr^{'}) \quad (8)$$

Therefore, when using the attack methods such as $Clustering(tr)$ or $Count(tr)$ on the synthesized trajectory, the final results are inconsistent with those on the original trajectory, that is, the individual identity cannot be correctly identified in the synthesized trajectory.

6.3. Evaluation of Privacy Preservation Degree

In the experiments of our TPP-DABD algorithm, the number of individual trajectory exchanges is set to 5 to 12, with an average value of 7.5. The number of trajectories involved in each exchange is 2 to 6, with an average value of 2.6. Therefore, we calculate the Error Rate of Trajectory Inference to be $ERoTI = (1 - \frac{1}{2.6^{7.5}}) \times 100\% = 99.92\%$.

The error rate of trajectory inference of each trajectory privacy-preserving algorithm is shown in Table 6.

Table 6. The $ERoTI$ of trajectory privacy-preserving algorithms.

ine Trajectory Privacy-Preserving Algorithms	$ERoTI$
ine $UIIDSM$	0.2958
$AIIDSM$	0.3066
$ARMM$	0.3802
$RMMUP$	0.7486
$DPCP$	0.8347
$STMPS$	0.9972
$TPP - DABD$	0.9992
ine	

By Table 6, the error rate of our TPP-DABD method is similar to that of STMPS, and is higher than the other methods.

6.4. Evaluation of Data Utility

We evaluated the utility of trajectory data mainly by the difference between the original and published trajectories in spatial and temporal distribution.

Many existing methods add false trajectory, trajectory offset, random location distribution and so on to protect trajectory data privacy. Nevertheless, in our method, all trajectory data came from the real trajectories, and retained the statistical characteristics of the original trajectory data to the greatest extent. There was almost no deviation on the statistical analysis between the results of our method and those using original data.

Taking "the top n locations in the region" as an example, we compared our method with the classical trajectory privacy-preserving algorithms by the metric of Statistical Feature Retention Rate (SFRR).

The SFRR of each algorithm is shown in Table 7.

Table 7. The SFRR of trajectory privacy-preserving algorithm.

ine Algorithms	**Top 10**	**Top15**	**Top 20**	**Top25**	**Top30**
ine $TPP-DABD$	100%	100%	100%	100%	100%
$UIIDSM$	5%	8%	10%	13%	18%
$AIIDSM$	11%	16%	27%	41%	60%
$ARMM$	17%	24%	31%	49%	64%
$RMMUP$	13%	21%	29%	37%	57%
$DPCP$	72%	79%	84%	89%	93%
$STMPS$	40%	46%	62%	67%	90%
ine					

Through the above analysis and comparisons, it can be concluded that our TPP-DABD algorithm had a stronger ability to retain the statistical characteristics of trajectory data. It was especially suitable for scenarios where trajectory data were statistically analyzed.

6.5. Evaluation of Time Complexity

Our experimental environment includes 1 server configured with 2 CPUs(Intel Xeon, E5-2620, 6 cores), memory of 128 GB, SSD of 2 TB, and $CentOS6.7\ (64bit)$. The algorithms are implemented by *node.js*.

We selects different sets of trajectory points and compares the operation times of the algorithms. The sizes of trajectory points are 50,000, 100,000, 200,000, 400,000, 800,000 respectively. The operation times of the above seven algorithms are shown in Figure 5, x-axis is operation time and y-axis is the number of trajectory location points.

From the above experiment results, we can see that the performance of these algorithms could be divided into three grades:

(1) The first grade was DPCP, UIIDSM, which had the lowest time cost. Because the algorithm only needed to consider a single trajectory point, the algorithm was simple.

(2) The second grade was AIIDSM, ARMM, RMMUP, and TPP-DABD. Their time costs were moderate, and they needed to calculate distributions of trajectory points (either overall or local).

(3) The third grade was STMPS. Due to the high complexity of semantic calculation, the algorithm took the longest time.

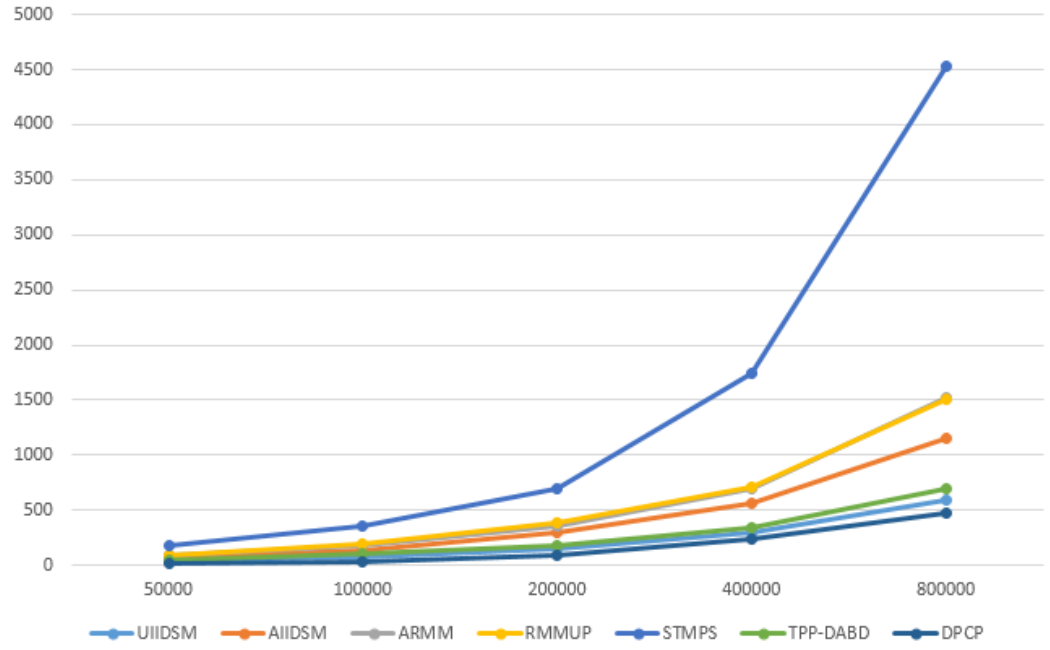

Figure 5. The operation times of the seven algorithms.

6.6. Performance Analysis of TPP-DABD

From the above analysis on the privacy preservation degree, data utility and the algorithm performance, we can see that the TPP-DABD was feasible for privacy-preserving trajectory data publishing.

(1) Effectiveness of TPP-DABD:

The TPP-DABD is mainly based on the secondary mixing of QI, so that the individual identity and QI are no longer in the one-to-one mapping relationship. The true mapping cannot be restored from the overall area, so that the trajectory cannot be associated with the individual.

The key of the TPP-DABD is to decompose the original trajectory into different QI segments. The more segments it is decomposed into, the better privacy it will achieve. In the experimental results of this paper, each trajectory was divided into 7.5 segments on average, and the inferential error rate was 99.92%. The TPP-DABD had a higher degree of privacy protection than those classical algorithms.

At the same time, TPP-DABD does not change the original geographical coordinates in the published trajectory points. The geographic statistics information based on the original trajectory can maintain 100% consistency with the sampling trajectory points, and the data utility is very high.

(2) Efficiency of TPP-DABD

Unlike the other privacy-preserving algorithms, the high data utility and privacy protection degree in our TPP-DABD were achieved without trading off too much time cost. The time complexity of the TPP-DABD was still acceptable for practical applications.

More importantly, TPP-DABD supports data processing by time slices. When the amount of data increases, the operation time of the algorithm increases linearly. At the same time, TPP-DABD supports parallel processing. In practical applications, the total operation time can be reduced by expanding the hardware resources.

7. Conclusions

This paper proposes a novel algorithm of Trajectory Privacy Preservation by Dynamic Anonymization with Bounded Distortion (TPP-DABD). Taking into account of the requirements of application scenarios of trajectory data, we define the evaluation metrics for

measuring the algorithm performance, and evaluate the performance of the algorithm based on real Guangzhou City Taxi trajectory data.

The major contribution of this paper is the proposed novel dynamic anonymization method based on localized trajectory mixing. The main advantage of the technique is that all published trajectory data are formed by mixing the real trajectories in a localized manner under minimum noise injection rather than globally as most existing methods do. This effectively guarantees a bounded distortion and enables to better retain the statistical characteristics of the data, resulting in a better data utility without sacrificing privacy.

Our experimental performance evaluation and comparisons with the existing methods show that our TPP-DABD algorithm performs comparably with the trajectory privacy-preserving method based on location semantics, and better than the existing methods based on static (global) anonymization such as i.i.d. sampling and random movement. In addition to the data utility of individual trajectories, our method also minimizes the distortion of the statistical features of trajectory data, to provide a high utility of the published data collectively for statistical analysis.

We notice that the quality of performance of TPP-DABD depends also on data distribution and the attacker's background knowledge. In the future, we will apply the differential privacy technique for trajectory privacy protection to against attacks with arbitrary background knowledge on the statistical query results.

Author Contributions: Conceptualization, Songyuan Li and Hong Shen; methodology, Songyuan Li and Hui Tian; software, Songyuan Li; validation, Songyuan Li, Hui Tian and Hong Shen; formal analysis, Songyuan Li; investigation, Songyuan Li; resources, Songyuan Li; data curation, Songyuan Li; writing—original draft preparation, Songyuan Li; writing—review and editing, Songyuan Li, Hui Tian, Hong Shen and Yingpeng Sang; visualization, Songyuan Li; supervision, Hong Shen; project administration, Yingpeng Sang; funding acquisition, Hong Shen and Yingpeng Sang. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Key-Area Research and Development Program of Guangdong Province, China (NO.2020B010164003), the National Key R & D Program of China Project grant number 2017YFB0203201 and the Science and Technology Program of Guangdong Province, China (No.2017A010101039).

Institutional Review Board Statement: No applicable.

Informed Consent Statement: No applicable.

Data Availability Statement: No applicable.

Acknowledgments: This work was supported by the Key-Area Research and Development Program of Guangdong Province, China (NO. 2020B010164003), the National Key R & D Program of China Project under Grant 2017YFB0203201 and the Science and Technology Program of Guangdong Province, China (No. 2017A010101039).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Huo, Z.; Meng, X.F. A survey of trajectory privacy-preserving techniques. *Jisuanji Xuebao Chin. J. Comput.* **2011**, *34*, 1820–1830. [CrossRef]
2. Sheng, G.; Ma, J.; Shi, W.; Zhan, G.; Cong, S. TrPF A Trajectory Privacy Preserving Framework for Participatory Sensing. *IEEE Trans. Inf. Forensics Secur.* **2013**, *8*, 874–887.
3. Bindschaedler, V.; Shokri, R. Synthesizing plausible privacy-preserving location traces. In Proceedings of the 2016 IEEE Symposium on Security and Privacy, San Jose, CA, USA, 26 May 2016; pp. 546–563.
4. Shokri, R.; Theodorakopoulos, G.; Danezis, G.; Hubaux, J.P.; Le Boudec, J.Y. Quantifying location privacy: The case of sporadic location exposure. In Proceedings of the International Symposium on Privacy Enhancing Technologies Symposium, Waterloo, ON, Canada, 27–29 July 2011; pp. 57–76.
5. Shokri, R.; Theodorakopoulos, G.; Troncoso, C.; Hubaux, J.P.; Boudec, J.Y.L. Protecting location privacy: optimal strategy against localization attacks. In Proceedings of the ACM Conference on Computer and Communications Security, Raleigh, NC, USA, 16–18 October 2012; pp. 617–627.

6. Chow, R.; Golle, P. Faking contextual data for fun, profit, and privacy. In Proceedings of the ACM Workshop on Privacy in the Electronic Society, Chicago, IL, USA, 9 November 2009.
7. Kumari, V.; Chakravarthy, S. Cooperative privacy game: a novel strategy for preserving privacy in data publishing. *Hum.-Centric Comput. Inf. Sci.* **2016**, *6*, 12. [CrossRef]
8. Kato, R.; Iwata, M.; Hara, T.; Suzuki, A.; Nishio, S. A dummy-based anonymization method based on user trajectory with pauses. In Proceedings of the International Conference on Advances in Geographic Information Systems, Redon Beach, CA, USA, 6–9 November 2012.
9. Pingley, A.; Nan, Z.; Fu, X.; Choi, H.A.; Wei, Z. Protection of query privacy for continuous location based services. In Proceedings of the IEEE INFOCOM, Shanghai, China, 10–15 April 2011.
10. Yu, W.; Xu, D.; Xiao, H.; Chao, Z.; Fan, L.; Xu, B.; Tang, S.J. L2P2: Location-aware Location Privacy Protection for Location-based Services. In Proceedings of the IEEE INFOCOM, Orlando, FL, USA, 25–30 March 2012.
11. Beresford, A.R.; Stajano, F. Mix Zones: User Privacy in Location aware Services. In Proceedings of the Second IEEE Conference on Annual Conference on Pervasive Computing and Communications Workshops, Orlando, FL, USA, 14–17 March 2004; pp. 127–131.
12. Machanavajjhala, A.; Gehrke, J.; Kifer, D.; Venkitasubramaniam, M. L diversity: Privacy beyond k anonymity. In Proceedings of the 22nd International Conference on Data Engineering (ICDE'06), Atlanta, GA, USA, 3–7 April 2006; p. 24.
13. Abul, O.; Bonchi, F.; Nanni, M. Never Walk Alone: Uncertainty for Anonymity in Moving Objects Databases. In Proceedings of the 2008 IEEE 24th International Conference on Data Engineering, Cancun, Mexico, 7–12 April 2008; pp. 376–385.
14. Bonchi, F.; Lakshmanan, L.V.S.; Wang, H. Trajectory anonymity in publishing personal mobility data. *Acm Sigkdd Explor. Newsl.* **2011**, *13*, 30–42. [CrossRef]
15. Gao, S.; Ma, J.; Sun, C.; Li, X. Balancing trajectory privacy and data utility using a personalized anonymization model. *J. Netw. Comput. Appl.* **2014**, *38*, 125–134. [CrossRef]
16. Shin, H.; Vaidya, J.; Atluri, V.; Choi, S. Ensuring Privacy and Security for LBS through Trajectory Partitioning. In Proceedings of the Eleventh International Conference on Mobile Data Management, Kansas City, MO, USA, 23–26 May 2010; pp. 224–226.
17. Mir, D.J.; Isaacman, S.; Caceres, R.; Martonosi, M.; Wright, R.N. DP-WHERE: Differentially private modeling of human mobility. In Proceedings of the IEEE International Conference on Big Data, Santa Clara, CA, USA, 6–9 October 2013.
18. Gursoy, M.E.; Liu, L.; Truex, S.; Yu, L. Differentially private and utility preserving publication of trajectory data. *IEEE Trans. Mob. Comput.* **2018**, *18*, 2315–2329. [CrossRef]
19. Zhao, X.; Pi, D.; Chen, J. Novel trajectory privacy-preserving method based on clustering using differential privacy. *Expert Syst. Appl.* **2020**, *149*, 113241. [CrossRef]
20. Li, Y.; Yang, D.; Hu, X. A differential privacy-based privacy-preserving data publishing algorithm for transit smart card data. *Transp. Res. Part Emerg. Technol.* **2020**, *115*, 102634. [CrossRef]
21. Proserpio, D.; Goldberg, S.; McSherry, F. Calibrating data to sensitivity in private data analysis: a platform for differentially-private analysis of weighted datasets. *Proc. Vldb Endow.* **2014**, *7*, 637–648. [CrossRef]
22. Dai, J.; Liang, H. A Method for the Trajectory Privacy Protection Based on the Segmented Fake Trajectory under Road Networks. In Proceedings of the 2015 2nd International Conference on Information Science and Control Engineering, Shanghai, China, 24–26 April 2015.
23. Sun, D.; Luo, Y.; Fan, G.; Guo, L.; Zheng, X. Privacy protection algorithm based on trajectory shape diversity. *J. Comput. Appl.* **2016**. *36*, 1544–1551.
24. Zhang, S.; Wang, G.; Liu, Q.; Liu, J. Trajectory Privacy Protection Method Based on Multi-Anonymizer. *J. Comput. Res. Dev.* **2019**, *56*, 576.
25. Zhu, W.; You, Q.; Yang, W.; Zhou, Q. Trajectory Privacy Preserving Based on Statistical Differential Privacy. *J. Comput. Res. Dev.* **2017**, *54*, 2825.
26. Li, S.; Shen, H.; Sang, Y.; Tian, H. An efficient method for privacy-preserving trajectory data publishing based on data partitioning. *J. Supercomput.* **2020**, *76*, 5276–5300. [CrossRef]
27. Li, S.; Shen, H.; Sang, Y. An efficient model and algorithm for privacy-preserving trajectory data publishing. In Proceedings of the International Conference on Parallel and Distributed Computing: Applications and Technologies, Jeju Island, Korea, 20–22 August 2018; pp. 240–249.
28. Liberti, L.; Lavor, C.; Maculan, N.; Mucherino, A. Euclidean distance geometry and applications. *Quant. Biol.* **2014**, *56*, 3–69. [CrossRef]
29. Wang, L.; Zhang, Y.; Feng, J. On the Euclidean distance of images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1334–1339. [CrossRef] [PubMed]
30. Krumm, J. Realistic Driving Trips For Location Privacy. In Proceedings of the Pervasive Computing, International Conference, Pervasive, Nara, Japan, 11–14 May 2016.
31. Petracca, G.; Marvel, L.M.; Swami, A.; Jaeger, T. Agility maneuvers to mitigate inference attacks on sensed location data. In Proceedings of the Military Communications Conference, Baltimore, MD, USA, 1–3 November 2016.