Electronic Research Archive

ERA, 33(11): 6631–6651.
DOI: 10.3934/era.2025293
Received: 06 August 2025
Revised: 13 October 2025
Accepted: 27 October 2025
Published: 12 November 2025

http://www.aimspress.com/journal/ERA

Research article

DP-FETC: a differentially private trajectory publishing method based on feature extraction and trajectory correlation

Bin Yue, Shuyu Li*, Anyu Liu and Xiongfei Li

School of Artificial Intelligence and Computer Science, Shaanxi Normal University, Xi'an 710119, China

* **Correspondence:** Email: lishuyu@snnu.edu.cn; Tel: +862985310161.

Abstract: With the widespread application of location-based services, how to effectively publish trajectories while preserving users' privacy has become a critical challenge. Existing privacy-preserving trajectory publishing methods often face issues such as missing in synthetic trajectories, low data utility, and privacy leakage of users' social relationships due to trajectory correlation. To address these issues, a differentially private trajectory publishing method based on feature extraction and trajectory correlation (DP-FETC) is proposed in this paper. The method is composed of two synergistic core algorithms. First, a trajectory synthesis algorithm (SynFE) employs adaptive grid discretization to extract key statistical features—including weighted location, origin-destination (OD), and length distributions—to generate high-fidelity synthetic trajectories that maintain high data utility. Second, to mitigate the risk of social relationship disclosure, an innovative correlated trajectory protection algorithm (PreOD) identifies highly correlated trajectories using a novel metric based on stay duration at points of interest. It then applies a targeted perturbation exclusively to the OD points of these high-risk trajectories. This strategy effectively obscures social links while minimizing the impact on overall data quality. Experimental results on two real-world datasets validate that the proposed method has good data utility while providing robust privacy guarantees.

Keywords: trajectory publishing; correlated trajectory; privacy preservation; differential privacy

1. Introduction

The proliferation of mobile crowd-sensing and location-based services has made spatiotemporal

trajectory data a vital asset for understanding human mobility and behavior, enhancing urban governance, and powering data-driven business intelligence, with applications from personalized navigation and intelligent traffic control to epidemic tracing, demonstrating substantial societal and economic impact [1–5]. However, these data inherently capture sensitive aspects of individual behavior such as travel routines, activity zones, and daily patterns. Serious privacy concerns, including identity re-identification, exposure of sensitive locations, and behavioral profiling [6], may be further raised if the data is released directly or used improperly. The challenge of unlocking the value of trajectory data while ensuring robust privacy preservation is a problem of growing importance in both academia and industry.

To address these pressing privacy concerns, a wide range of trajectory publishing techniques have been developed, including anonymity-based generalization, cryptographic obfuscation methods, and formal privacy frameworks such as differential privacy [7]. Among them, differential privacy (DP) provides a mathematically rigorous framework for privacy-preserving data publishing and has been extensively studied for trajectory synthesis [8,9]. However, many existing differentially private trajectory publishing techniques often struggle to achieve the fidelity of key distributional features of original trajectories [3,10–12]. For instance, many methods fail to accurately capture intricate statistical patterns inherent in real trajectories—such as fine-grained location visit frequencies, precise origin-destination (OD) distribution, or variations in trajectory length. This deficiency can significantly degrade the performance of downstream applications like traffic flow modeling which relies on accurate OD patterns, or urban mobility analysis which requires detailed visit frequencies [11,13]. More critically, current approaches often overlook inter-trajectory correlations. Human behaviors inherently exhibit social patterns, leading to strong correlations in users' trajectories when individuals attend shared events or frequent common destinations. For instance, individuals participating in the same private gathering may generate trajectories with strong spatiotemporal similarity, which, if unprotected, could be exploited by an adversary to infer sensitive social connections or co-attendance at specific locations, thereby posing a serious risk of correlation-based privacy leakage [14,15]. More critically, most existing methods treat each trajectory as an isolated entity, thereby overlooking an important dimension of privacy risk—the inter-trajectory correlation that arises from shared mobility patterns [16,17]. Such correlations enable adversaries to infer implicit social relationships through spatiotemporal co-occurrence analysis (e.g., users repeatedly appearing together at the same private or semi-private locations) [14,18]. This type of inference represents a higher-order privacy leakage, extending beyond individual re-identification. A few studies have attempted to address this problem [16,17,19]; however, they typically rely on complex optimization models that impair data quality or adopt global perturbation strategies that lack targeted protection, resulting in excessive utility degradation.

To address the above challenges, this paper proposes DP-FETC, a framework for differentially private trajectory publishing that simultaneously preserves individual mobility features and protects social-interaction privacy. The main contributions of this work are summarized as follows:

(1) We develop a dual-layer privacy framework that jointly safeguards individual-level trajectory statistics and inter-user correlations. By integrating the SynFE and PreOD algorithms, DP-FETC ensures high-fidelity trajectory synthesis while introducing a correlation-aware protection mechanism to mitigate social relationship inference—an aspect largely overlooked in prior studies.

(2) We design a semantically enhanced correlation metric and a targeted OD perturbation strategy to protect social-interaction privacy. The proposed correlation metric incorporates users' stay duration

at points of interest (POIs), capturing meaningful co-presence patterns beyond geometric similarity. Based on this metric, the PreOD algorithm selectively perturbs the OD points of high-risk trajectories, achieving effective yet low-distortion privacy protection.

(3) We conduct comprehensive experiments on two real-world trajectory datasets, GeoLife and Porto, to evaluate the proposed framework across diverse mobility contexts. The results demonstrate that DP-FETC effectively preserves both trajectory fidelity and privacy protection under different urban mobility patterns, confirming its scalability, robustness, and practicality for real-world deployment.

The remainder of the paper is organized as follows. In Section 2, related work is discussed. Section 3 provides a review of preliminary knowledge about kernel density estimation and differential privacy. Section 4 details the proposed DP-FETC method. In Section 5, the results of experiments conducted on two real datasets are analyzed. Finally, Section 6 concludes the paper.

2. Related work

The imperative to balance data utility with individual privacy in an increasingly data-driven world has fueled significant research into privacy-preserving trajectory publishing. While existing approaches have explored anonymization, generalization, and cryptographic methods [12,18,20,21], differential privacy (DP) [22] has emerged as a promising technique, offering a robust and quantifiable framework for privacy preservation independent of adversarial auxiliary knowledge.

Some differentially private methods for trajectory data mainly focused on location-level perturbation, where noise was directly added to a user's locations. For instance, Andrés et al. [10] added Laplace noise to raw trajectories, while Li et al. [23] developed methods for adaptively distributing noise using spatial indexes. Besides, semantic concepts have been utilized for enhancing utility by researchers. To better preserve semantic information, Du et al. [24] designed a hierarchical graphical model (HGM) to capture movement patterns between semantic categories, enabling the synthesis of trajectories with good semantic and geographic utility. Zhu et al. [25] proposed location-discriminative geo-indistinguishability (LDGI), which enhanced geo-indistinguishability by incorporating location-discriminative sensitivity. LDGI provided tailored privacy protection based on location importance, optimizing the utility-privacy trade-off. However, a major limitation of these location-level techniques is their inherent tendency to disrupt higher-order statistical properties of the trajectory data, such as traffic volume distributions and complex mobility motifs. This significantly compromised the utility of the synthetic data.

To address these limitations, feature-based synthesis methods have been developed. These methods extract essential statistical features from original trajectories, privatize them under differential privacy, and recreate synthetic data from noisy models. Ghane et al. [26] introduced the trajectory generative mechanism (TGM), a graph-based generative model that encodes movement statistics and adaptively adds noise while synthesizing trajectory data. Sun et al. [27] proposed a privacy-preserving and utility-enhancing framework for trajectory synthesization (PUTS), which leverages road network structures and path-level information, such as travel times and route choices, to enhance synthetic trajectory realism and utility; it extracts path-based features and synthesizes trajectories respecting network constraints under differential privacy. Sun et al. [28] proposed a novel solution for synthesizing private and realistic trajectories (SPRT), which synthesizes more realistic trajectories under differential privacy by integrating geographic structures, utilizing a geography-aware grid, and applying movable constraints to better preserve individual-level mobility patterns. Du et al. [29]

proposed a locally differentially private trajectory synthesis method (LDPTrace) that achieves local differential privacy by perturbing key movement patterns extracted from user trajectories, such as length and transitions, and synthesizes realistic trajectories from these aggregated perturbed patterns without relying on external knowledge. While feature-based methods marked a considerable advancement in utility preservation, some of the following issues remain: Coarse spatial discretization can obscure fine-grained mobility details, and the optimal selection and combination of task-relevant features under stringent privacy budgets remains a non-trivial problem. More critically, these methods predominantly focused on safeguarding individual privacy, largely overlooking the substantial privacy threats emanating from inter-user correlations inherent in shared spatio-temporal experiences.

In addition to individual privacy, inter-user correlations based on shared routines or experiences bring further privacy threats. Trajectory co-occurrence or co-similarity can be exploited by attackers to infer social relationships as well as group activities. Yu et al. [16] proposed the method for correlated trajectory publication with differential privacy (CTP), which frames correlation protection as a constrained optimization problem. It quantifies correlation using cell visit probability vectors derived from an adaptive grid and employs a differentially private particle swarm optimization algorithm to minimize inter-trajectory similarity before synthesis. Building on this, Wu et al. [17] introduced a trajectory correlation privacy-preserving mechanism (TCPP), prioritizing data availability while protecting multi-user correlation. The proposed method first uses a Kalman filter to predict user trajectories, then applies a personalized privacy budget based on time and distance to safeguard correlation privacy. Yuan et al. [19] broadened the focus to semantic correlation with the semantic correlation trajectory privacy-preserving mechanism (SCTP). This mechanism leverages a hidden Markov model (HMM) to model semantic state transitions and allocates privacy budgets according to the frequency of semantic information, effectively protecting user interests and activity types from inference. However, existing related research still exhibits several common issues: many methods rely on complex predictive or optimization models, which introduces uncertainty and may compromise data quality due to the models' inherent errors; their methods for quantifying correlation are difficult to balance, either being too simplistic to capture complex associations or increasing modeling overhead by introducing complex dimensions like semantics; and their protection mechanisms often apply global perturbations to the data, which can lead to unnecessary distortion and fail to optimally balance privacy and utility.

3. Preliminaries

3.1. Kernel density estimation (KDE)

Kernel density estimation (KDE) [30,31] is a non-parametric statistical method for estimating the underlying probability density function of a dataset without assuming a specific parametric model. It is particularly useful for modeling the spatial distribution of locations. Given a set of n two-dimensional locations $\{p_1,...,p_n\}$, the KDE at a location p is computed as:

$$\hat{f}(p)=\frac{1}{n}\sum_{i=1}^{n}K_H(p-p_i) \tag{1}$$

where K_H is the scaled kernel function—a smoothing function used to distribute each location's

influence. The parameter H is a bandwidth matrix, which is often simplified to a single scalar h for isotropic smoothing. KDE is applied in trajectory synthesis due to its flexibility and ability to capture complex, non-uniform spatial patterns.

3.2. Differential privacy (DP)

Definition 1 (ε-differential privacy [22]). Let M denote a randomized algorithm, and S represents an arbitrary set of possible outputs of M. For any two neighboring data sets D_1 and D_2, which differ by only one record, the algorithm is said to be ε-differentially private if:

$$\Pr\left[M\left(D_1\right)\in S\right]\le e^{\varepsilon}\times\Pr\left[M\left(D_2\right)\in S\right] \tag{2}$$

where $\Pr[\cdot]$ denotes the probability of an event.

Definition 2 (Laplace mechanism [22]). For a query function $f:D\to R^d$ with sensitivity Δf, if the output result satisfies Eq (2), the algorithm is said to provide ε-differential privacy:

$$M=f\left(D\right)+Laplace(\Delta f/\varepsilon) \tag{3}$$

where the noise level is proportional to the global sensitivity Δf and inversely proportional to the privacy parameter ε.

Theorem 1 (sequential combination [22]). Suppose there is a set of random algorithms $\left\{M_1,M_2,...,M_n\right\}$, where each of $M_i\left(1\le i\le n\right)$ satisfies ε_i-differential privacy on the dataset. Then the set of M_i sequence privacy mechanisms provides $\left(\sum_{i=1}^{n}\varepsilon_i\right)$-differential privacy.

Theorem 2 (parallel combination [22]). Suppose the dataset D can be divided into a series of independent and non-overlapping subsets $\left\{D_1,D_2,...,D_n\right\}$ and there is a set of random algorithms $\left\{M_1,M_2,...,M_n\right\}$. If each privacy mechanism satisfies differential privacy in $D_i\left(1\le i\le n\right)$, then the set of randomized algorithms can achieve $\max\left\{\varepsilon_i\right\}$-differential privacy on the dataset D.

4. Design of DP-FETC

To address prevalent issues, including missing critical features in synthetic data and the risk of exposing social relationships due to trajectory similarity, a differentially private trajectory publishing method based on feature extraction and trajectory correlation (DP-FETC) is proposed, and the design of the DP-FETC method is presented in this section.

4.1. The framework of the DP-FETC

The DP-FETC method applies adaptive spatial discretization to the real trajectory dataset to facilitate fine-grained partitioning of activity areas, making the statistical data features between the real trajectories and the synthetic trajectories more similar. Furthermore, privacy preservation is only applied to the OD (origin and destination) of highly correlated trajectories, which reduces the information loss of trajectory data and improves data utility. The incentive is to simulate the behavior pattern of the real trajectories while protecting sensitive locations with high correlation. The synthetic

trajectories are more aligned with the real ones, reducing the leakage risk of social relationships and improving the accuracy of the trajectory dataset to be released.

The DP-FETC method is built upon two core components: the trajectory feature extraction and synthesis module and the OD preservation for correlated trajectories module, as shown in Figure 1: a trajectory synthesis module and a correlated trajectory OD protection module, whose core functionalities are implemented by the SynFE and PreOD algorithms, respectively. These two algorithms jointly achieve privacy-preserving trajectory publishing by ensuring both the fidelity of synthetic trajectories and the protection of correlation-based privacy risks.

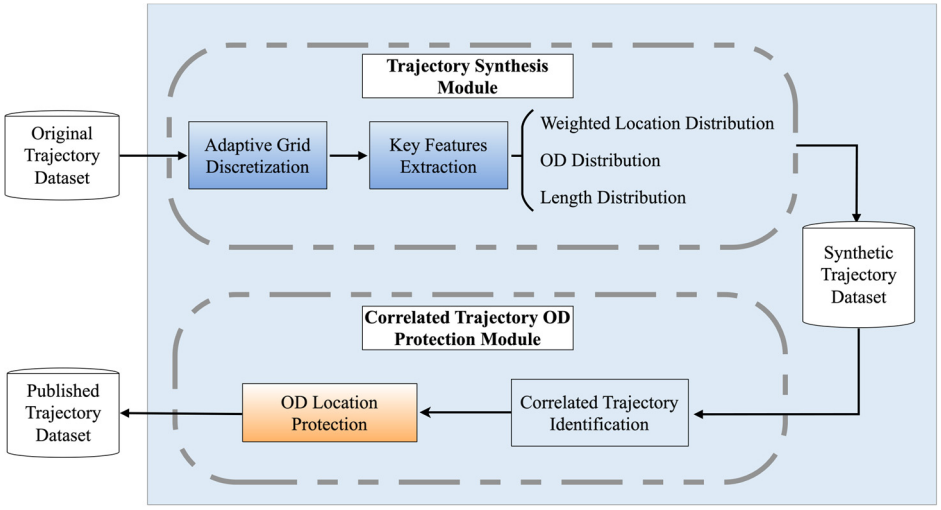

Figure 1. Framework of the DP-FETC method. The modules highlighted with a color gradient are the core stages for achieving privacy protection. Among them, the adaptive grid discretization and key features extraction modules, indicated by a blue gradient, consume the privacy budget by injecting noise to satisfy differential privacy. Meanwhile, the OD location protection module, indicated by an orange gradient, utilizes the post-processing property of differential privacy to enhance protection without consuming additional budget.

The first one is dedicated to extracting key features of real trajectories and generating the synthetic ones, and a trajectory synthesis algorithm based on feature extraction (SynFE) is designed. The SynFE algorithm begins by dividing the activity area of real trajectories using an adaptive grid strategy. It then extracts essential features, including weighted location distribution, OD distribution and length distribution. These features, in conjunction with kernel density estimation, are used to generate synthetic trajectories that statistically approximate the real ones.

The other one quantifies trajectory correlation and preserves OD for correlated trajectories. Correspondingly, a privacy-preserving algorithm for the origin and destination (PreOD) is presented. Relying on the output of SynFE, the PreOD algorithm first identifies correlated trajectory pairs that might reveal social relationships of users according to semantic and spatial-temporal attributes. Then it adds noise to the OD of highly correlated trajectories.

Together, these two components work in tandem to release synthetic trajectories that maintain the core characteristics of the original dataset under the protection of differential privacy.

4.2. SynFE algorithm

The SynFE algorithm is designed to generate synthetic trajectories that statistically approximate the original ones while adhering to differential privacy. First, we discretize the original trajectory, finely partition the top coarse grid and bottom fine grid by considering density and distance, and then extract key features of the trajectory.

4.2.1. Adaptive grid discretization

To effectively capture local features and accommodate the non-uniform spatial distribution characteristic of trajectory data, the SynFE algorithm employs an adaptive grid discretization strategy. It begins by determining the activity area, which is the smallest rectangular area containing all original trajectories. Then the activity area is adaptively partitioned into grids based on the density and distance distribution of locations. Generally, a coarse-grained grid is a larger grid with low density, while a fine-grained grid is a smaller grid with high density. When the movement distance between locations is small in a coarse-grained grid, it is necessary to divide this grid into several fine-grained grids to more accurately capture subtle changes in the trajectory. On the contrary, if the movement distance between locations in the grid is large, the number of grids should be reduced to more efficiently reflect the macroscopic movement trend of the trajectory.

Thus, the activity area is divided into a coarse uniform grid of $B * B$ cells; any of these coarse grid cells can be further divided into fine grid cells of different sizes based on the density and distance information. And G represents the whole adaptive grid. The value of B is determined as follows:

$$B = max\left(10, \frac{1}{4}\left\lceil \frac{N\varepsilon_1}{10} \right\rceil\right) \tag{4}$$

where N is the number of locations in the original dataset, and ε_1 is the privacy budget allocated to the grid discretization. This formula provides a heuristic, inspired by prior work in differentially private spatial data publishing, for adaptively setting the initial grid resolution based on the dataset size N and the privacy budget ε_1. The constants 4 and 10 are empirical parameters chosen to manage the fundamental trade-off between grid resolution and the magnitude of injected noise. This balance is critical: the grid must be fine enough to capture key mobility patterns, yet not so fine that the noise required for privacy protection excessively degrades the overall data utility.

The noisy location count $\tilde{m}_i$ of the coarse-grained grid cell G_i is defined as follows:

$$\tilde{m}_i = m_i + Laplace(1/\varepsilon_1) \tag{5}$$

where m_i is the real location count of G_i.

Subsequently, a coarse-grained grid cell G_i can be further divided into $C_i * C_i$ fine-grained grid cells:

$$C_i = \left\lceil \beta l(G_i)/s(G_i) \right\rceil \tag{6}$$

where $l(G_i)$ is the total length of the trajectories in the grid cell G_i, $s(G_i)$ is the area of G_i, $l(G_i)/s(G_i)$ represents the grid density of G_i, and β is the granularity weight to adjust the value

range of C_i.

The density of trajectories within each coarse grid cell determines the fineness of this subdivision, as depicted in Figure 2, allowing areas with more trajectory activity to be modeled with greater detail, while less active areas remain coarser. This two-layer grid structure enables a more nuanced representation of varying activity intensities across different regions while maintaining privacy.

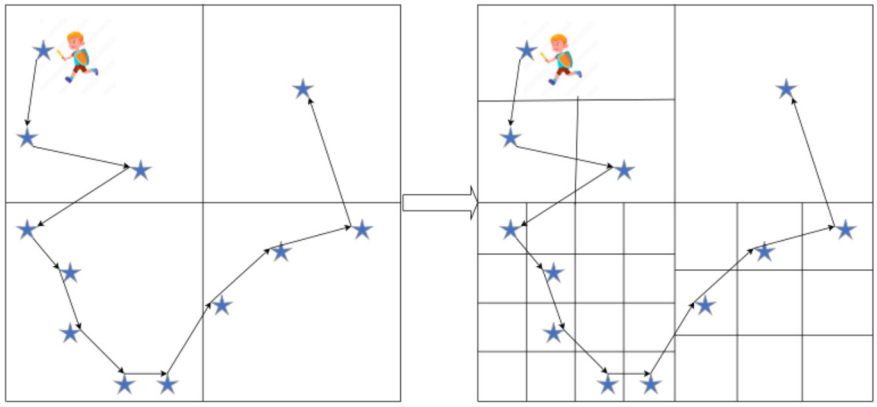

Figure 2. An example of adaptive grid discretization.

4.2.2. Trajectory feature extraction

Once the adaptive grid is established, the SynFE algorithm extracts and privatizes three key trajectory features under differential privacy: weighted location distribution, OD distribution, and length distribution. These features collectively characterize both macroscopic and microscopic behavior patterns.

(1) Weighted location distribution

This feature aims to model the overall spatial distribution of activity. While locations are generally distributed uniformly within each fine-grained grid cell, the number of locations allocated to each cell is influenced by itself and its neighboring cells. For a fine-grained grid cell G_{ij} inside a coarse-grained cell G_i, the weighted noisy location count $\tilde{m}_{ij}$ allocated to G_{ij} is defined as follows:

$$\tilde{m}_{ij} = \tilde{m}_i \left(\omega \frac{A_{ij}}{A_i} + (1-\omega) \frac{m'_{ij}}{nb_{ij}} \right) \tag{7}$$

where A_{ij} and A_i are the area of G_{ij} and G_i, respectively; and $\omega (0 \le \omega \le 1)$ is the weighting factor, setting $\omega = 0.5$, which means to give the same influence weight to G_{ij} and its neighbors. $\tilde{m}_i$ is the noisy location count of G_i. The key local density terms, m'_{ij} and nb_{ij}, are defined below.

The term m'_{ij} is the noisy location counts of G_{ij}:

$$m'_{ij} = m_{ij} + Laplace(1/\varepsilon_2) \tag{8}$$

where m_{ij} is the real location count of G_{ij}, and ε_2 is the privacy budget.

The term nb_{ij} is the sum of the noisy location counts of G_{ij} and all of its neighboring cells:

$$nb_{ij} = \sum_{Neighbor_{ij}} m'_{ik} \tag{9}$$

where, m'_{ik} is the noisy location counts of neighboring cell G_{ik} of G_{ij}.

Once the weighted location count for each fine-grained grid cell is determined. The SynFE algorithm adopts a triangular location sampling strategy: each grid cell is divided into triangles using its centroid and adjacent cell vertices, and the number of synthetic locations generated within these triangles is proportional to their area.

(2) OD distribution

To model the OD (origin-destination) pattern of users, the SynFE algorithm introduces a virtual origin location p_{vo} and destination location p_{vd}, which can be conceptually linked to all the grid cells. Supposing a trajectory $Traj_i$, its location is p_t which belongs to grid cell G_x at timestamp t. The transition probability of movement from G_x to its adjacent grid cell G_y is defined as follows:

$$Pr(OD_{xy}) = \begin{cases} Pr(p_{t+1} \in G_y \mid p_t \in G_x), \ if \ G_y \in Neighbor_{G_x} \\ 0, \qquad\qquad\qquad\qquad \text{otherwise} \end{cases} \tag{10}$$

where $Neighbor_{G_x}$ is the set of neighboring cells of G_x. For an origin transition (from p_{vo} to the origin location of $Traj_i$) or a destination transition (from the destination location of $Traj_i$ to p_{vd}), its transition probability is always equal to 1.

Thus, for a trajectory $Traj_i$, a transition probability matrix M_i can be constructed. To preserve privacy, Laplace noise is added to the items of M_i using the privacy budget ε_3, resulting in a noisy matrix $\tilde{M}_i$.

(3) Length distribution

The distribution of trajectory lengths is a fundamental statistical feature, as it captures the scale of mobility by distinguishing short trips from long-distance travel. Accurately preserving this distribution is therefore crucial for both the realism of the synthetic data and the utility of downstream applications, such as traffic flow analysis, which rely on faithful aggregate travel statistics.

Given the trajectory $Traj_i$ and its OD grid cells, the trajectory length of $Traj_i$ is also an important feature. To conserve the privacy budget, the SynFE algorithm derives this distribution from the noisy transition matrix $\tilde{M}_i$. The length distribution of $Traj_i$ is defined as follows:

$$\Pr(Traj_i, l) = \frac{\Pr(Traj_i \mid len = l)}{\sum_{d=l_{\min}}^{l_{max}} \Pr(Traj_i \mid len = d)} \tag{11}$$

where $\Pr(Traj_i, l)$ represents the probability of the trajectory $Traj_i$ having length l. $l_{\min}$ is the minimum length, which can be calculated by the Dijkstra algorithm under the condition of treating the whole grid as a graph. $l_{\min}$ is the maximum length and $l_{max} = \alpha l_{\min}$, where α is an upper bound factor.

4.2.3. Trajectory synthesis via kernel density estimation

To realistically generate synthetic locations from weighted location uniform distribution, the SynFE algorithm applies kernel density estimation (KDE), specifically using a Laplace-based kernel

defined in polar coordinates. The probability density function (PDF) is given as follows:

$$\sigma(p-\hat{p})=\sigma(r,\theta)=\frac{\exp(-r/h)}{2\pi h} \tag{12}$$

where p and $\hat{p}$ are the sampled location and synthetic location, respectively, $r=\|p-\hat{p}\|$, θ is the angle between p and $\hat{p}$, and h is the smoothing factor.

In order to obtain a differentially private kernel for a grid cell G_i, it is necessary to adjust the kernel function in each cell. Therefore, the smoothing parameter h_i for G_i is determined by:

$$h_i=\frac{\|G_i\|}{\varepsilon_2} \tag{13}$$

where $\|G_i\|$ is the maximum distance between any two locations in G_i and ε_2 is the privacy budget used in the weighted location uniform distribution. Therefore, by checking the probability ratio between $\sigma(0,\theta)$ and $\sigma(\|G_i\|,\theta)$, it can be easily proved that the kernel function satisfies differential privacy.

Once the KDE is set up, for a sampled location p, independently calculating r and θ from the calibrated kernel, then latitude and longitude of the synthetic location $\hat{p}$ can be determined: $\hat{p}_{lat}=p_{lat}+rcos\theta$ and $\hat{p}_{lon}=p_{lon}+rsin\theta$.

The main process of the SynFE algorithm is given as follows.

Algorithm 1: SynFE

Input: Original trajectory dataset D_{raw}, privacy budget $\varepsilon=\varepsilon_1+\varepsilon_2+\varepsilon_3$

Output: Synthetic trajectory dataset D_{syn}

1. Initialize $D_{syn}=\varnothing$;
2. Partition grid cells according to Eqs (4) and (6);
3. For each $Traj_i$ in D_{raw}
4. Calculate OD distribution according to Eq (10);
5. Calculate noisy transition probability matrix $\tilde{M}_i$;
6. Calculate the length distribution according to Eq (11) and choose l with maximum probability as the trajectory length;
7. Set the OD locations of the synthetic trajectory $Traj'_i$ as the OD locations of $Traj_i$;
8. For $j=2$ to $l-1$ do
9. Sample a location p from the weighted location distribution;
10. Calculate r and θ via kernel density estimation;
11. Generate a synthetic location $\hat{p}$;
12. Add $\hat{p}$ to $Traj'_i$;
13. end for
14. Add $Traj'_i$ to D_{syn};
15. end for
16. Return D_{syn}.

4.3. PreOD algorithm

Considering that the correlated trajectories may reveal social relationships between users, the PreOD algorithm is designed to obscure the correlated trajectories to a certain extent, so that attackers with certain background knowledge cannot uniquely identify a specific trajectory.

4.3.1. Quantifying trajectory correlation

Traditional trajectory similarity metrics, such as dynamic time warping (DTW) and the Hausdorff distance, primarily capture geometric path similarity but overlook semantic context. Consequently, they may fail to distinguish meaningful social interactions from incidental encounters. For instance, two trajectories following the same major highway may appear highly similar without implying any social relationship, whereas two users repeatedly meeting at a specific point of interest (POI), such as a café or residence, reveal a much stronger social connection.

To address this limitation, we introduce a correlation metric that incorporates the duration of stay at shared POIs as a key weighting factor. The underlying intuition is that longer co-location durations at common POIs are indicative of stronger social ties. This semantically grounded metric enables the PreOD algorithm to focus perturbation on trajectory pairs with the highest likelihood of leaking social relationship information, thereby enhancing privacy protection while minimizing unnecessary utility loss.

To identify correlated trajectories that might inadvertently disclose social relationships, the PreOD algorithm begins by grouping trajectories with the same or geographically proximate OD locations. Within each group, the correlation between any two trajectories of different users is calculated. The POI stay durations are used to weigh the significance of shared locations as defined in Eqs (14) and (15).

Supposing a synthetic trajectory $Traj_i$ belonging to a user A, and the POI set contained in the $Traj_i$ is $POI_{Ai} = \{poi_1, poi_2, ..., poi_n\}$, a Euclidean distance matrix Dis_{Ai} can be constructed, where an element $dis_{ij} \in Dis_{Ai}$ holds the distance between poi_i and poi_j $(1 \le i, j \le n)$.

The stay time weight w_{Ai} for poi_i is defined as follows:

$$w_{Ai} = \frac{stime_{Ai}}{\sum_{j=1}^{n} stime_{Aj}} \tag{14}$$

Thus, the distance metric of $Traj_i$ of user A is calculated as follows:

$$dismet_{Ai} = \sum_{i=1}^{n} \sum_{j=1}^{n} dis_{ij} * weight_{Ai} * weight_{Aj} / 2 \quad i \ne j \tag{15}$$

Similarly, the distance metric of $dismet_{Bj}$ for a synthetic trajectory $Traj_j$ of a user B can also be determined.

Finally, the trajectory correlation $Corr_{ij}$ between $Traj_i$ and $Traj_j$ is calculated as:

$$Corr_{ij} = \frac{min(dismet_{Ai}, dismet_{Bj})}{max(dismet_{Ai}, dismet_{Bj})} \tag{16}$$

Utilizing this correlation index, the PreOD algorithm employs a Top-k ranking approach to

identify and select the k trajectory pairs with the highest correlation scores for each pair of users to form the correlated dataset D_{rel}. In our experiments, we set *k = 1* to focus on protecting the strongest potential social link while minimizing the impact on overall data utility.

4.3.2. OD location protection for correlated trajectories

Since OD locations of a trajectory often indicate user intent and can be privacy-sensitive, the PreOD algorithm specifically protects these OD locations of trajectories in the correlated dataset D_{rel}. Because synthetic trajectories may not be perfectly accurate or smooth, Gaussian noise is added to the OD locations of correlated trajectories to better emulate real trajectories.

Let $p_{io} = < p_{io}.lat, p_{io}.lon, p_{io}.t>$ and $p_{id} = < p_{id}.lat, p_{id}.lon, p_{id}.t>$ be the origin location and the destination location of the trajectory $Traj_i$ in the D_{rel}, with latitude, longitude, and timestamp attributes, respectively. We randomly generate Gaussian noise rate α_{io}, α_{id} and angle β_{io}, β_{id}, and the latitude and longitude of synthetic OD locations $\hat{p}_{io}$ and $\hat{p}_{id}$ are calculated as follows:

$$\begin{aligned} \hat{p}_{io}.lat &= p_{io}.lat + \alpha_{io} \cdot sin(\beta_{io}) \\ \hat{p}_{io}.lon &= p_{io}.lon + \alpha_{io} \cdot cos(\beta_{io}) \\ \hat{p}_{id}.lat &= p_{id}.lat + \alpha_{id} \cdot sin(\beta_{id}) \\ \hat{p}_{id}.lon &= p_{id}.lon + \alpha_{id} \cdot cos(\beta_{id}) \end{aligned} \tag{17}$$

The synthetic OD locations are then validated for their utility by using spatial constraints. Taking the origin location as an example, the spatial constraint is given as follows:

$$\sqrt{(\hat{p}_{io}.lat - p_{io}.lat)^2 + (\hat{p}_{io}.lon - p_{io}.lon)^2} \le \delta \tag{18}$$

where δ represents the spatial threshold, and the spatial constraint of the synthetic destination location is similar to the above formula.

A synthetic OD location is considered usable only if it satisfies the above spatial constraint, otherwise the noise injection process is repeated until the constraint is satisfied.

The main process of the PreOD algorithm is given as follows.

Algorithm 2: PreOD

Input: synthetic trajectory dataset D_{syn}, spatial threshold δ

Output: released trajectory dataset D_{pub}

1. Group trajectories with the same or geographically proximate OD locations;
2. Initialize the correlated sets $D_{rel} = \varnothing$ and $D_{pub} = D_{syn}$;
3. for each trajectory group do
4. for any $Traj_i$ of user A and $Traj_j$ of user B
5. Calculate correlation $Corr_{ij}$ according to Eq (16);
6. end for
7. Find the trajectory pair ($Traj_k$, $Traj_m$) with the maximum correlation value between users A and B;

8. Add $Traj_k$ and $Traj_m$ to D_{rel};
9. Remove $Traj_k$ and $Traj_m$ from D_{pub};
10. end for
11. for each $Traj_l$ in D_{rel}
12. repeat
13. Generate the synthetic OD locations of $Traj_l$ according to Eq (17);
14. until spatial constraint is satisfied
15. Add $Traj_l$ with the synthetic OD locations to D_{pub};
16. end for
17. Return D_{pub}.

4.4. Complexity and privacy analysis

This section provides an analysis of the time complexity of the DP-FETC method and a proof of its adherence to ε-differential privacy.

Time Complexity. The complexity of DP-FETC is determined by its two main components. The SynFE algorithm, which involves grid partitioning and feature extraction, has a near-linear complexity of $O(N \cdot L)$, where N is the number of trajectories and L is their average length. The dominant computational cost, however, resides in the PreOD algorithm, which requires a pairwise comparison to quantify the correlation between all user trajectories. This step has a quadratic complexity of $O(N^2)$. Therefore, the overall time complexity of the DP-FETC method is $O(N^2)$.

Privacy Analysis. The DP-FETC method rigorously satisfies ε-differential privacy. The privacy budget ε is entirely consumed within the SynFE algorithm across three sequential stages: adaptive grid discretization (consuming ε_1), weighted location distribution extraction (consuming ε_2), and OD distribution extraction (consuming ε_3). Each stage is rendered differentially private using the Laplace mechanism. By the sequential composition theorem [22], the SynFE algorithm as a whole provides ($\varepsilon = \varepsilon_1 + \varepsilon_2 + \varepsilon_3$)-DP. Subsequently, the PreOD algorithm operates exclusively on the already-privatized synthetic dataset produced by SynFE. Since PreOD does not access the original data, its operations qualify as post-processing. According to the post-processing property of differential privacy, this stage incurs no additional privacy cost. Thus, the entire DP-FETC framework is formally ε-differentially private.

5. Experiments

5.1. Experimental settings

We implemented the proposed method in Python programming language (version 3.9), and conducted experiments on a machine with an Intel (R) i7-11700 CPU @ 2.50 GHz, 16G RAM, 512 SSD + 1T hardware configuration and a 64-bit Windows 10 operating system.

Our experimental study utilized two real datasets, GeoLife [32] and Porto [33]. GeoLife is a trajectory dataset of 182 users collected by the MSRA Geolife project from April 2007 to August 2012. Each trajectory includes a series of locations with timestamp, latitude, and longitude attributes, containing 17621 trajectory data. Porto is a trajectory dataset of 442 taxis in Porto city

from January 7, 2013, to June 30, 2014. It has approximately 1.57 million trajectory data, with an interval of 15 seconds between every two locations. Table 1 outlines the statistics of the two datasets.

Table 1. Statistics of datasets.

Data Set	City	Sample Size
GeoLife	Beijing	17,621
Porto	Porto	1,570,000

To ensure data quality, comparability, and suitability for privacy-preserving trajectory publishing, both datasets were carefully preprocessed before use. First, anomalous trajectories with fewer than five points were removed. To standardize temporal resolution, all trajectories were resampled to a uniform 15-second interval, where sparse trajectories were linearly interpolated and dense trajectories were downsampled. For the Porto dataset, additional preprocessing was applied to eliminate vehicle-specific artifacts and make the trajectories more representative of general urban mobility behaviors. Specifically, a representative subregion within the dense urban core was selected to reduce large-scale driving patterns, and trajectories were refined by removing very short trips while truncating overly long ones into shorter, continuous segments. These steps yielded smaller-scale, fine-grained trajectory segments that effectively simulate pedestrian-like movement within realistic city environments. After preprocessing, both datasets provide consistent, high-quality mobility traces suitable for evaluating the fidelity and privacy protection of synthetic trajectory publishing.

In our experimental evaluation, the total privacy budget ε was allocated equally across the three privacy-preserving stages of the SynFE algorithm. Specifically, the budgets for adaptive grid discretization (ε_1), weighted location distribution (ε_2), and OD distribution (ε_3) were set such that $\varepsilon_1 = \varepsilon_2 = \varepsilon_3 = \varepsilon/3$. This equal allocation strategy was chosen because it provides a fair and unbiased baseline for protection, particularly in the absence of prior knowledge that would justify a different weighting scheme.

5.2. Experimental results

To evaluate the performance of the proposed method, we have employed Jensen-Shannon divergence (JSD) as the metric for data utility and mutual information (MI) as the metric for privacy preservation.

JSD is used to measure the statistical similarity between the distribution of the original dataset and synthetic dataset. This statistical fidelity is a crucial indicator of practical utility. A lower JSD value signifies a smaller discrepancy between the two distributions, which in turn indicates that the statistical patterns of the synthetic trajectories closely resemble the real ones. Such a resemblance implies that the synthesis process has preserved the essential data characteristics with minimal distortion. Therefore, a lower JSD score serves as a direct and robust indicator of higher data utility. JSD is defined as:

$$JS = \frac{1}{2} KL\left(P, \frac{P+Q}{2}\right) + \frac{1}{2} KL\left(Q, \frac{P+Q}{2}\right) \tag{19}$$

where P and Q are the probability distributions of the original dataset and the synthetic dataset, respectively. KL denotes the Kullback-Leibler divergence.

We evaluate JSD on three key trajectory characteristics: spatial location, temporal duration, and spatial distance. $Pg(G_i)$ is the spatial location and denotes the probability of a grid cell G_i being visited, reflecting the popularity of a region and the rough location distribution of a trajectory. $Pt(Traj_i)$ is the temporal duration distribution of trajectory $Traj_i$. $Ps(Traj_i)$ is the distance distribution of trajectory $Traj_i$, and we use the sum of Manhattan distances between consecutive locations to calculate distance.

MI is used to quantify the amount of information that synthetic trajectories reveal about the original trajectories. The smaller the mutual information between a pair of original trajectory and synthetic trajectory values, the lower the degree of information overlap between them, and the less likely it is to leak personal trajectory privacy.

Given a real trajectory $Traj_i$ and its corresponding synthetic trajectory $Traj'_i$, the MI value is calculated based on their joint and marginal probability distributions. The MI for a pair of original trajectory and synthetic trajectory value is defined as follows:

$$MI(Traj_i, Traj'_i) = \sum_t p\left(p_t, \hat{p}_t\right) log \frac{p\left(p_t, \hat{p}_t\right)}{p\left(p_t\right) p(\hat{p}_t)} \tag{20}$$

where p_t and $\hat{p}_t$ are the locations of $Traj_i$ and $Traj'_i$ at timestamp t, $p(p_t, \hat{p}_t)$ is the joint probability, and $p(p_t)$ and $p(\hat{p}_t)$ are the marginal probabilities.

Given an original trajectory set D_{raw} and a synthetic trajectory set D_{pub}, we define the average MI as:

$$MI(D_{raw}, D_{pub}) = \frac{1}{|D_{raw}|} \sum_{i=1}^{|D_{raw}|} MI(Traj_i, Traj'_i) \tag{21}$$

where $|D_{raw}|$ is the size of D_{raw}, and $|D_{raw}| = |D_{pub}|$.

We compare the DP-FETC method with the following four methods, which are representative of different approaches to trajectory synthesis and privacy:

FTS [34]: A feature-preserving trajectory synthesis method that uses the gene partitioning strategy to decompose trajectories.

TSG [35]: A two-stage GAN-based method for large-scale GPS trajectory generation using a modified deep GAN, in which the generator adopts encoder/decoder architecture.

DP-TrajGAN [36]: A privacy-preserving trajectory synthesis method that incorporates LSTMs into a GAN framework and adopts a partially observable Markov decision process for privacy budget allocation. It is selected here as a state-of-the-art, deep learning-based method that also provides differential privacy, serving as our primary DP-based benchmark.

LSTM [37]: An LSTM-based method that learns user behavior patterns from a real dataset, and combines discriminative and generative models to learn the joint probability distribution for synthetic mobility traffic generation.

Two groups of experiments are conducted. The first group of experiments tests the difference between the real trajectory dataset and the synthetic trajectory dataset generated by the DP-FETC method, from the relationship view of the frequency with normalized latitude and longitude. The second group of experiments compares the JSD values and MI values of the above five methods. The experiments adopt a five-fold cross-validation strategy to test the accuracy of the above methods and take the average as the result.

(1) Comparison of the real dataset and the dataset synthesized by DP-FETC

To visually assess how well the proposed method preserves fundamental trajectory features, we compare the frequency distribution of the normalized latitude and longitude between the real trajectories and those synthesized by the DP-FETC method. Figures 3 and 4 present these comparisons for the GeoLife and Porto datasets, respectively.

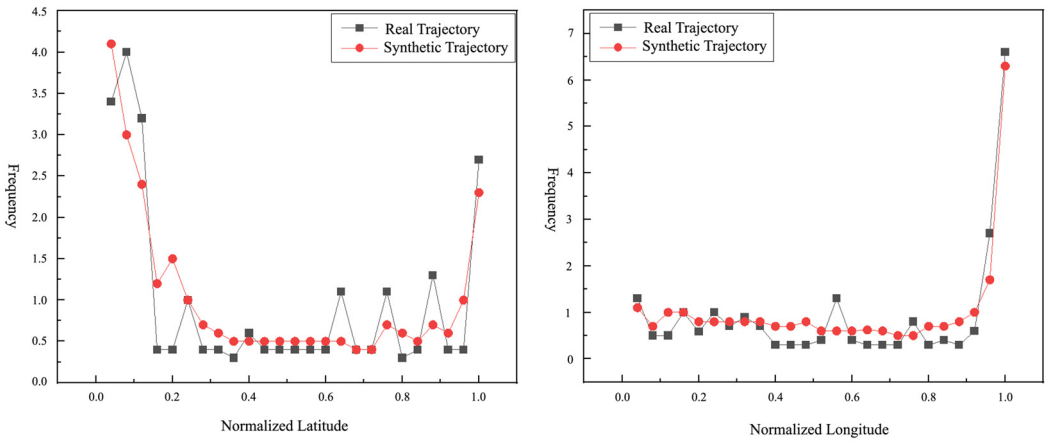

Figure 3. Comparison of frequency distribution (GeoLife).

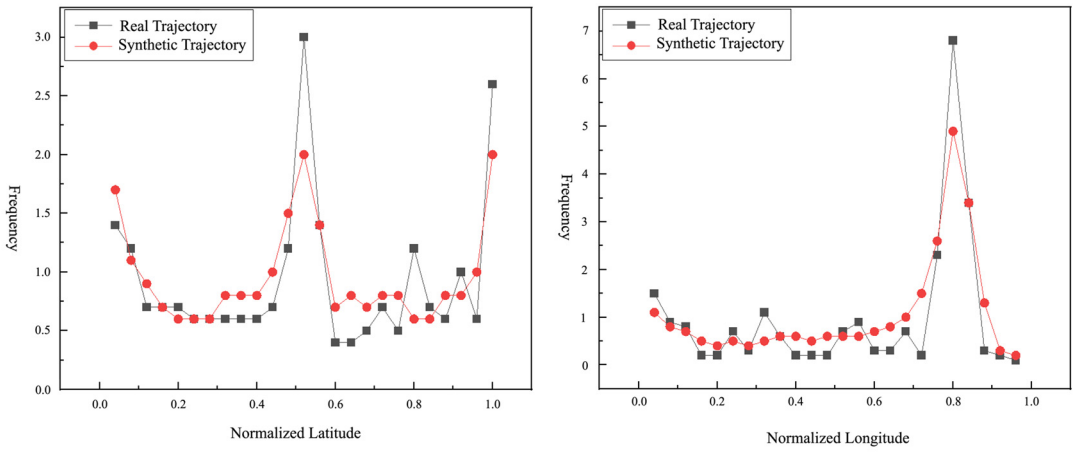

Figure 4. Comparison of frequency distribution (Porto).

As observed from Figures 3 and 4, the distributions of the normalized latitude and longitude for trajectories synthesized by the DP-FETC method closely follow those of the real trajectories on both datasets. This demonstrates that the proposed method, through its adaptive grid discretization and feature extraction mechanism, can effectively capture and replicate the primary spatial characteristics of the original movement patterns.

(2) Comparison of JSD and MI

We evaluate the data utility of the synthesized trajectories using JSD across the three features: spatial location, temporal duration, and spatial distance. Lower JSD values are preferable. Figure 5 shows the JSD values for the above methods on the GeoLife and Porto datasets.

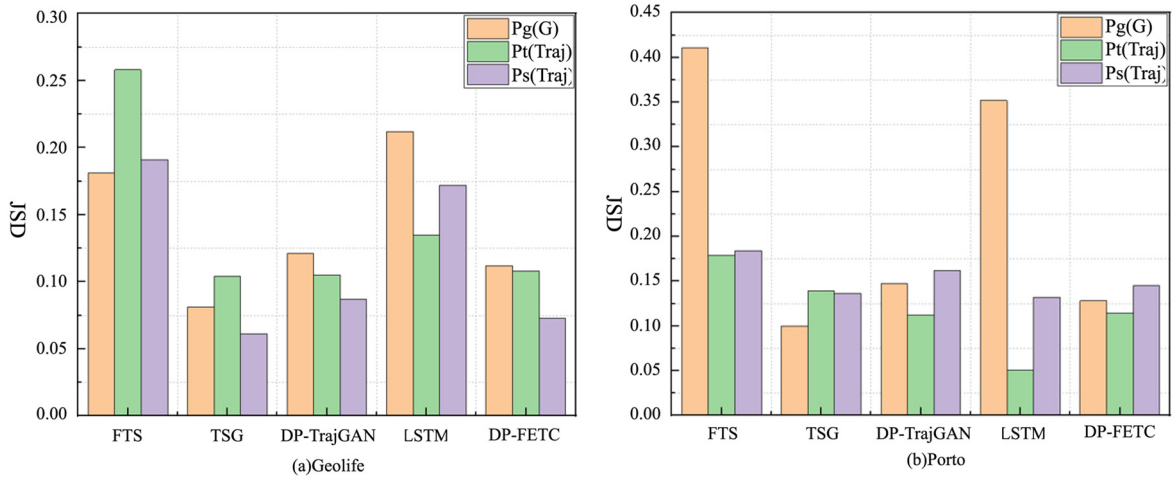

Figure 5. JSD comparison across three features.

It can be seen from Figure 5(a), the TSG method achieves the best score, while the proposed method also achieves competitive JSD scores across all three features on the GeoLife dataset. Its performance is close to TSG, particularly for spatial distance, and it is marginally higher but still demonstrates better utility for spatial location and temporal duration. The proposed method is generally better than FTS and LSTM. This indicates that the proposed method effectively preserves these diverse statistical properties.

It can also be seen from Figure 5(b), similar trends are observed on the Porto dataset. The above results demonstrate that the proposed method effectively balances the noise injection required for privacy with the need to maintain statistical fidelity.

To assess the privacy preservation offered by the DP-FETC method, particularly its ability to obfuscate the relationship between original and synthetic trajectories, MI values are calculated and lower MI values are preferred. Figure 6 shows the MI values for the above methods on the GeoLife and Porto datasets.

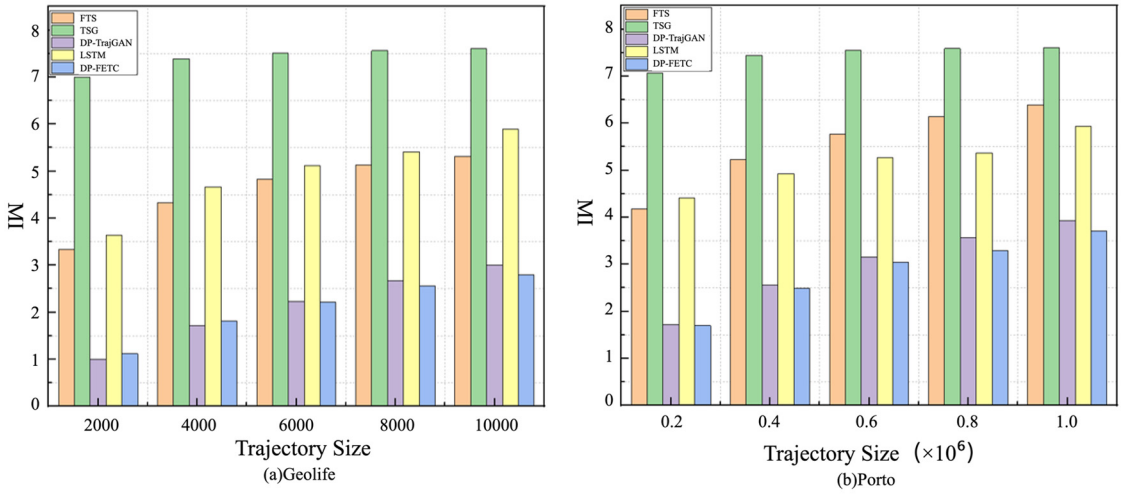

Figure 6. MI comparison.

It can be observed that MI value increases as the number of trajectories increases. As shown in Figure 6(a), the DP-FETC method generally achieves lower or comparable MI values compared to the DP-TrajGAN method on the GeoLife dataset. When the number of trajectories is 2000 or 4000, the MI value of the proposed method is slightly higher than that of the DP-TrajGAN method. However, as the number of trajectories increases, the MI value of the proposed method is equal to or slightly lower than that of the DP-TrajGAN method. The proposed method is more efficient in identifying correlated trajectories and weakens the correlation for trajectory datasets with a larger size. The proposed method outperforms the TSG, FTS, and LSTM methods.

As shown in Figure 6(b), the DP-FETC method demonstrates the lowest MI values among all the methods for the test dataset with different sizes. This highlights its performance in decorrelating synthetic trajectories from their originals, thereby offering better privacy assurances against re-identification or linkage attacks.

The experimental results on two real datasets indicate that the DP-FETC method strikes an effective balance between data utility and privacy preservation. In terms of data utility (JSD), DP-FETC is highly competitive, closely matching or outperforming several baselines by accurately preserving key features of trajectories. In terms of privacy (MI), the DP-FETC method generally exhibits better performance, particularly for larger datasets. This comprehensive performance suggests that the proposed method is a promising approach for publishing trajectory data that is both useful for analysis and robustly protects individual privacy.

6. Conclusions

A differentially private trajectory publishing method named DP-FETC is proposed in this paper. It is designed to preserve key mobility features while mitigating correlation-based privacy risks. Its effectiveness stems from its two core components: the SynFE algorithm ensuring data fidelity through feature extraction, and the PreOD algorithm providing robust privacy by selectively perturbing the OD locations of highly correlated trajectories. Experiments on two real datasets validate that this approach achieves better data utility while offering privacy preservation against several existing methods. Our future work will extend this framework by exploring more complex correlation patterns and investigate adaptive privacy budget allocation strategy for diverse analytics tasks.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research was funded by the Natural Science Basic Research Program of Shaanxi Province for Young Scientists under Grant 2023-JC-QN-0762.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. D. Wang, T. Miwa, T. Morikawa, Big trajectory data mining: A survey of methods, applications, and services, *Sensors*, **20** (2020), 4571. https://doi.org/10.3390/s20164571
2. X. F. Xiao, C. H. Li, X. J. Wang, A. P. Zeng, Personalized tourism recommendation model based on temporal multilayer sequential neural network, *Sci. Rep.*, **15** (2025), 382. https://doi.org/10.1038/s41598-024-84581-z
3. X. Wang, Z. Jerome, Z. Wang, C. Zhang, S. Shen, V. V. Kumar, et al., Traffic light optimization with low penetration rate vehicle trajectory data, *Nat. Commun.*, **15** (2024), 1306. https://doi.org/10.1038/s41467-024-45427-4
4. B. Benreguia, H. Moumen, M. A. Merzoug, Tracking COVID-19 by tracking infectious trajectories, *IEEE Access*, **8** (2020), 145242–145255. https://doi.org/10.1109/ACCESS.2020.3015002
5. S. Wang, Z. Bao, J. S. Culpepper, G. Cong, A survey on trajectory data management, analytics, and learning, *ACM Comput. Surv.*, **54** (2021), 1–36. https://doi.org/10.1145/3440207
6. F. Jin, W. Hua, M. Francia, P. Chao, M. E. Orlowska, X. Zhou, A survey and experimental study on privacy-preserving trajectory data publishing, *IEEE Trans. Knowl. Data Eng.*, **35** (2023), 5577–5596. https://doi.org/10.1109/TKDE.2022.3174204
7. Y. Zhan, H. Haddadi, A. Kyllo, A. Mashhadi, Privacy-aware human mobility prediction via adversarial networks, in *2022 2nd International Workshop on Cyber-Physical-Human System Design and Implementation (CPHS)*, (2022), 7–12. https://doi.org/10.1109/CPHS56133.2022.9804533
8. À. Miranda-Pascual, P. Guerra-Balboa, J. Parra-Arnau, J. Forné, T. Strufe, SoK: Differentially private publication of trajectory data, *Proc. Priv. Enhancing Technol.*, **2023** (2023), 438–457. https://doi.org/10.56553/popets-2023-0065
9. R. Bhadani, A survey on differential privacy for spatiotemporal data in transportation research, preprint, arXiv:2407.15868. https://doi.org/10.48550/arXiv.2407.15868
10. M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, C. Palamidessi, Geo-indistinguishability: Differential privacy for location-based systems, in *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, (2013), 901–914. https://doi.org/10.1145/2508859.2516735
11. N. Wang, M. Kankanhalli, Dptraj-pm: Differentially private trajectory synthesis using prefix tree and markov process, preprint, arXiv:2404.14106. https://doi.org/10.48550/arXiv.2404.14106
12. Y. Jiang, Y. Wu, S. Zhang, J. J. Q. Yu, Fedvae: Trajectory privacy preserving based on federated variational autoencoder, in *2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*, (2023), 1–7. https://doi.org/10.1109/VTC2023-Fall60731.2023.10333794
13. E. Buchholz, A. Abuadbba, S. Wang, S. Nepal, S. S. Kanhere, Reconstruction attack on differential private trajectory protection mechanisms, in *Proceedings of the 38th Annual Computer Security Applications Conference*, (2022), 279–292. https://doi.org/10.1145/3564625.3564628
14. R. Shokri, G. Theodorakopoulos, C. Troncoso, Privacy games along location traces: A game-theoretic framework for optimizing location privacy, *ACM Trans. Priv. Secur.*, **19** (2016), 1–31. https://doi.org/10.1145/3009908
15. J. Long, T. Chen, G. Ye, K. Zheng, Q. V. H. Nguyen, H. Yin, Physical trajectory inference attack and defense in decentralized poi recommendation, in *Proceedings of the ACM Web Conference 2024*, (2024), 3379–3387. https://doi.org/10.1145/3589334.3645410

16. Y. Yu, H. Zhu, M. Xie, CTP: Correlated trajectory publication with differential privacy, in *2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS)*, (2021), 128–133. https://doi.org/10.1109/ICCCS52626.2021.9449263
17. L. Wu, C. Qin, Z. Xu, Y. Guan, R. Lu, TCPP: Achieving privacy-preserving trajectory correlation with differential privacy, *IEEE Trans. Inf. Forensics Secur.*, **18** (2023), 4006–4020. https://doi.org/10.1109/TIFS.2023.3290486
18. Z. Zheng, Z. Li, J. Li, H. Jiang, T. Li, B. Guo, Utility-aware and privacy-preserving trajectory synthesis model that resists social relationship privacy attacks, *ACM Trans. Intell. Syst. Technol.*, **13** (2022), 1–28. https://doi.org/10.1145/3495160
19. H. Yuan, L. Wu, L. Xu, L. Ban, H. Wang, Y. Su, L. Ban, H. Wang, Y. Su, et al., SCTP: Achieving semantic correlation trajectory privacy-preserving with differential privacy*, IEEE Trans. Veh. Technol.*, **74** (2025), 5856–5870. https://doi.org/10.1109/TVT.2024.3505200
20. L. Kuang, W. Shi, X. Chen, J. Zhang, H. Liao, A location semantic privacy protection model based on spatial influence, *Sci. Rep.*, **15** (2025), 15227. https://doi.org/10.1038/s41598-025-88553-9
21. D. Liu, G. Yu, Y. Ding, Z. Zhong, C. Wang, Privacy preserving multi-party computation with secret sharing for trajectory prediction in vanets, *IEEE Trans. Veh. Technol.*, **2024** (2024). https://doi.org/10.1109/TVT.2024.3432614
22. C. Dwork, Differential privacy: A survey of results, in *International Conference on Theory and Applications of Models of Computation*, (2008), 1–19. https://doi.org/10.1007/978-3-540-79228-4_1
23. S. Li, Y. Geng, Y. Li, A Differentially private hybrid decomposition algorithm based on quad-tree*, Comput. Secur.*, **109** (2021), 102384. https://doi.org/10.1016/j.cose.2021.102384
24. X. Du, H. Zhu, Y. Zheng, R. Lu, F. Wang, H. Li, A semantic-preserving scheme to trajectory synthesis using differential privacy, *IEEE Internet Things J.*, **10** (2023), 13784–13797. https://doi.org/10.1109/JIOT.2023.3262964
25. Y. Zhu, Y. Hong, Q. Xue, X. Lan, Y. Zhang, Y. Xiang, LDGI: Location-discriminative geo-indistinguishability for location privacy, *IEEE Trans. Knowl. Data Eng.*, **37** (2024). https://doi.org/10.1109/TKDE.2024.3522320
26. S. Ghane, L. Kulik, K. Ramamohanarao, TGM: A generative mechanism for publishing trajectories with differential privacy, *IEEE Internet Things J.*, **7** (2020), 2611–2621. https://doi.org/ 10.1109/JIOT.2019.2943719
27. X. Sun, Q. Ye, H. Hu, J. Duan, Q. Xue, T. Wo, et al., Puts: Privacy-preserving and utility-enhancing framework for trajectory synthesization, *IEEE Trans. Knowl. Data Eng.*, **36** (2024), 296–310. https://doi.org/10.1109/TKDE.2023.3288154
28. X. Sun, Q. Ye, H. Hu, Y. Wang, K. Huang, T. Wo, Synthesising realistic trajectory data with differential privacy, *IEEE Trans. Intell. Transp. Syst.*, **24** (2023), 5502–5515. https://doi.org/10.1109/TITS.2023.3241290
29. Y. Du, Y. Hu, Z. Zhang, Z. Fang, L. Chen, B. Zheng, et al., Ldptrace: Locally differentially private trajectory synthesis, preprint, arXiv:2302.06180. https://doi.org/10.48550/arXiv.2302.06180
30. G. S. Watson, Fiducial inference, *J. R. Stat. Soc. Ser. B*, **39** (1977), 319–321.
31. R. A. Davis, K. S. Lii, D. N. Politis, Remarks on some nonparametric estimates of a density function, in *Selected Works of Murray Rosenblatt*, Springer, (2011), 95–100. https://doi.org/10.1007/978-1-4419-8339-8_13

32. Y. Zheng, X. Xie, W. Y. Ma, GeoLife: A collaborative social networking service among user, location and trajectory, *IEEE Data Eng. Bull*., **33** (2010), 32–39.
33. L. Moreira-Matias, J. Gama, M. Ferreira, J. Mendes-Moreira, L. Damas, Predicting taxi-passenger demand using streaming data, *IEEE Trans. Intell. Transp. Syst*., **14** (2013), 1393–1402. https://doi.org/10.1109/TITS.2013.2262376
34. J. Li, W. Chen, A. Liu, Z. Li, L. Zhao, FTS: A feature-preserving trajectory synthesis model, *Geoinformatica*, **22** (2018), 49–70. https://doi.org/10.1007/s10707-017-0301-6
35. X. Wang, X. Liu, Z. Lu, H. Yang, Large scale GPS trajectory generation using map based on two stage GAN, *J. Data Sci*., **19** (2021), 126–141. https://doi.org/10.6339/21-JDS1004
36. J. Zhang, Q. Huang, Y. Huang, Q. Ding, P. W. Tsai, Dp-trajgan: A privacy-aware trajectory generation model with differential privacy, *Future Gener. Comput. Syst*., **142** (2023), 25–40. https://doi.org/10.1016/j.future.2022.12.027
37. V. Kulkarni, B. Garbinato, Generating synthetic mobility traffic using RNNs, in *Proceedings of the 1st Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery*, (2017), 1–4. https://doi.org/10.1145/3149808.3149809

AIMS Press