| | |
|---|---|
| Title | Spatial-Temporal Similarity for Trajectories with Location Noise and Sporadic Sampling |
| Authors | Li, Guanyao; Hung, Chih-Chieh; Liu, Mengyun; Pan, Linfei; Peng, Wen-Chih; Chan, Gary Shueng Han |
| Source | Proceedings - International Conference on Data Engineering, v. 2021-April, April 2021, article number 9458932, p. 1224-1235 |
| Version | Accepted Version |
| DOI | 10.1109/ICDE51399.2021.00110 |
| Publisher | IEEE |
| Copyright | © 2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. |

# Spatial-Temporal Similarity for Trajectories with Location Noise and Sporadic Sampling

Guanyao Li*, Chih-Chieh Hung†, Mengyun Liu*, Linfei Pan*, Wen-Chih Peng‡, S.-H. Gary Chan*
*Department of Computer Science and Engineering, The Hong Kong University of Science and Technology
†Department of Computer Science and Engineering, National Chung Hsing university
‡Department of Computer Science, National Yang Ming Chiao Tung University
gliaw@cse.ust.hk, smalloshin@email.nchu.edu.tw, {amylmy,lpanaf}@cse.ust.hk, wcpeng@g2.nctu.edu.tw, gchan@cse.ust.hk

***Abstract*—With the rapid advances and the penetration of the Internet of Things and sensors, a massive amount of trajectory data, given by discrete locations at certain timestamps, have been extracted or collected. Knowing the similarity between trajectories is fundamental to understanding their spatial-temporal correlation, with direct and far-reaching applications in contact tracing, companion detection, personalized marketing, etc. In this work, we consider the general and realistic sensing scenario that the locations of the trajectories may be noisy, and that these trajectories are sporadically sampled with randomness and asynchrony from the underlying continuous paths. Most of the prior work on trajectory similarity has not sufficiently considered the temporal dimension, or the issues of location noise and sporadic sampling, while others have limitations of strong assumptions such as a fixed known speed of users or the availability of a large amount of training data.**

**We propose a novel and effective spatial-temporal measure termed STS (Spatial-Temporal Similarity) to evaluate the spatial-temporal overlap between any two trajectories. In order to account for the location noise and sporadic sampling, STS models each location in a trajectory as an observable outcome drawn from a probability distribution. With that, it efficiently reduces the need for training data by estimating a personalized spatial-temporal probability distribution of the object position from its own trajectory. Based on that, it subsequently computes the co-location probability and hence derives the similarity of any two trajectories. We have conducted extensive experiments to evaluate STS using real large-scale indoor (mall) and outdoor (taxi) datasets. Our results show that STS is substantially more accurate and robust than the state-of-the-art approaches, with an improvement of $63\%$ on precision and $85\%$ on mean rank.**

***Index Terms*—spatial-temporal trajectory similarity, spatial-temporal data management, trajectory mining**

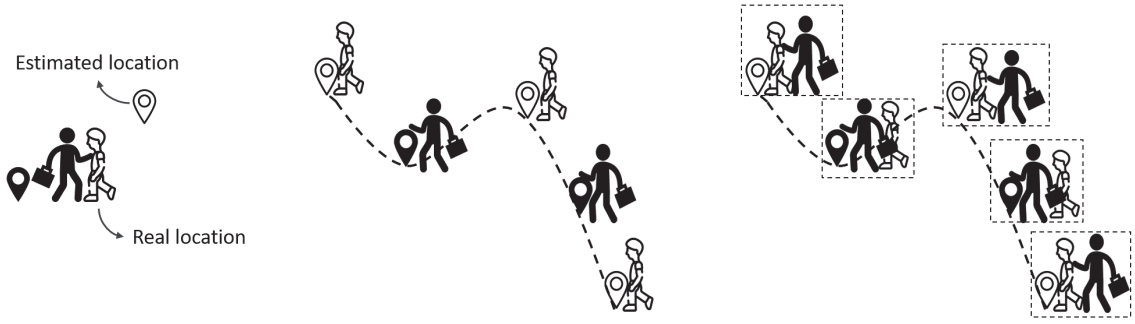
## I. Introduction

In recent years we have witnessed the rapid advances and the penetration of the Internet of Things (IoT) and sensing devices. Objects (e.g., users or their devices) may now know their locations based on signals such as GPS, WiFi, Bluetooth, video, etc. Besides, they may also leave their trails when using different service platforms such as call detail records (CDR) in telecommunication, smart cards in public transportation, mobile payments (e.g., banks, Alipay, Apple Pay, etc.), and O2O apps (e.g., bicycle-sharing, ride-hailing, etc.). Consequently, many sensing systems nowadays have extracted a massive amount of *trajectories*, each of which is a sequence of *positions* indicated by spatial locations and their corresponding timestamps sampled from a continuous *path* of the object.

A co-location occurs when two object paths are at the same spatial grid concurrently, the so-called spatial-temporal (S-T) overlap. The S-T similarity between two trajectories measures their level of co-location, i.e., how much the two trajectories overlap in the spatial and temporal dimensions. Such a similarity measure has many important applications. One is to match the trajectories of the same object in different sensing systems [1] [2]. As an object may leave multiple trajectories in different sensing systems, these trajectories need to be correlated for applications such as contact tracing [1], multimodal sensing [3], user re-identification [4], criminal investigation [5], etc. Furthermore, spatial-temporal similarity measure is also fundamental to companion detection for viral marketing, promotion and advertising [6]–[10], etc.

We propose and study a novel and effective measure to evaluate spatial-temporal similarity for trajectories. The problem is challenging, as we consider the following general and realistic scenarios on the uncertain trajectories due to the issues of location noise and sporadic sampling:

- *Location noise:* The process of location extraction and estimation is fundamentally noisy [11], [12]. As a result, two physically co-located objects may not appear so in their trajectories, or vice versa. We illustrate in Figure 1(a) two co-located people whose estimated locations may be separated by a rather wide margin. Due to location noise (estimation error), it would no longer be sound to measure the spatial-temporal similarity for trajectories by simply comparing their locations directly.
- *Sporadic sampling:* Due to the nature of sensing and beaconing devices, object paths are often sampled sporadically, i.e., trajectories are asynchronous with their locations collected randomly and independently with possibly time-varying heterogeneous rates. As a result, positions in an object path are not always observed in its trajectory and two objects walking together may not share overlapping trajectories, making it difficult to measure the spatial-temporal similarity simply based on co-locations in their trajectories. We show an example in Figure 1(b), where the trajectories of two people walking together are sampled at different times. Even without




(a) Issue of location noise. (b) Issue of sporadic sampling. (c) Co-location probability estimation.

Fig. 1. Illustrations of issues and intuition: (a) Co-located people may be estimated at different locations because of estimation noise. (b) People walking along the same route (the dotted line) have different sampled locations because of sporadic and asynchronous sampling. (c) The spatial-temporal similarity of two trajectories can be measured by estimating the co-location probabilities (the dotted box) at the locations of the two merged trajectories.

any location noise, their trajectories share no common positions. Moreover, the trajectories may be very sparse and irregular in some sensing systems (such as CDR, mobile payments, and tap in/out using smart cards in transportation systems), making the similarity comparison for all the collected trajectories even more challenging.

Much of the prior study on trajectory similarity considered only the spatial dimension [13]–[17]. They can hardly be extended to the spatial-temporal case we consider here. For the other works on spatial-temporal similarity, some form a pair-wise alignment for positions of two trajectories and compare their distance, without considering the general realistic issues of location noise and sporadic sampling [18]–[21]. Others are based on strong assumptions such as a fixed known speed of users or the availability of a large amount of data for model training [1], [2], [22], [23].

The co-location probability of two objects is defined as the probability that their paths fall in the same spatial grid at the same time. Two trajectories with high spatial-temporal similarity should have high co-location probabilities in their merged trajectory. We illustrate this in Figure 1(c), where the trajectories of two people are merged, and their co-location probability (the dotted box) at each of the positions is computed.

Given the above observation, we propose STS, a novel and effective **S**patial-**T**emporal **S**imilarity measure for trajectories with location noise and sporadic sampling. Instead of just calculating the point-wise distance for two trajectories (such as [13]–[15], [18]–[21]), STS compares similarity of trajectories by evaluating their co-location probability at different timestamps. The STS of two trajectories, as derived by the average of these co-location probabilities, would be high for co-locating objects, and low otherwise. To this end, STS first estimates the probability distribution of object position even if the location at that time is not observed in its trajectory. Then it compares the spatial-temporal similarity of any two trajectories by evaluating their co-location probabilities at the timestamps in their merged trajectory.

To tackle the challenges of location noise and sporadic sampling, STS exploits a *personalized transition probability estimator* to estimate the probability distribution of object locations at any arbitrary time (i.e., *spatial-temporal probability*) based on its speed. While most existing approaches for transition probability estimation focus on deriving a universal distribution for all objects by considering the transitions in the spatial space [24], [25], STS considers the transition probability estimation in both spatial and temporal dimensions for individual objects, i.e., transiting from one location at time $t_1$ to another location at time $t_2$. The transition probability estimated in STS is hence *personalized* and *spatial-temporal dependent*.

In particular, STS uses a kernel density estimation (KDE) to model the personalized speed probability distribution of an object from its own trajectory, and uses its speed distribution to denote the object transition probability between two locations in a given time interval. Our speed distribution estimation is fundamentally different from prior works [1] [26] [22] [23], which learn the universal speed distribution for all objects based on the assumptions that the form of the probability distribution is known and a large amount of training data is available.

Based on the spatial-temporal probability of objects, their *co-location probability* at any time can then be computed. The *spatial-temporal similarity* between any two trajectories is hence given by the average co-location probability in the merged trajectory.

We conducted an extensive experimental study to evaluate STS and compare it with the state-of-the-art approaches. Two real large-scale datasets — an outdoor taxi trajectory dataset collected in a city and an indoor pedestrian trajectory dataset collected in a large shopping mall — were used for evaluation. Our experimental results demonstrate its effectiveness and accuracy to measure spatial-temporal similarity in both outdoor and indoor scenarios (with an improvement of $63\%$ on precision and $85\%$ on mean rank). The results also demonstrate the robustness of STS against location noise and

sporadic data sampling.

The remainder of this paper is organized as follows. We present related works in Section II, followed by the preliminaries in Section III. We present the spatial-temporal probability estimation in Section IV. After that, we introduce the co-location probability and formulate the STS measure in Section V. We then discuss the experimental settings and results in Section VI. Finally, we conclude in Section VII.

## II. Related Works

Trajectory similarity measures have been previously proposed and studied in many works [27]. Most existing metrics consider the spatial closeness of trajectories, such as DTW [13], EDR [14], and ERP [28], [29]. In these works, a position in a trajectory is matched with another trajectory. Approaches such as dynamic programming is used to find the best alignment for these positions. After that, the distance between the positions in these trajectories is computed as their similarity. While these early works are impressive, they only consider the observed positions in a trajectory. Since trajectories are sporadically sampled, co-location in two paths may not be observable in the two trajectories. As a result, pairwise matching for positions in two trajectories fails to reflect the closeness between two underlying paths. To tackle the issues, t2vec [16] exploits a sequence-to-sequence model to learn the latent representation to compute similarity. The above works consider spatial similarity only, and they cannot be extended to our spatial-temporal case directly.

Some other works consider trajectory similarity in both the spatial and temporal dimensions. However, most of them are usually based on some strong assumptions or manually predefined parameters. For example, Fréchet distance [30] uses the largest distance between locations of two trajectoies at the same time to measure their similarity. Since locations in two trajectories are not always sampled concurrently, and noisy location is always far away from a trajectory, Fréchet distance is very sensitive to noise and sporadic sampling. STLIP [31] uses the in-between polylines distance and defines a temporal distance to measure spatial-temporal similarity. Nevertheless, it can only be applied to trajectories with two-dimensional spatial data. WGM [19] and another work [20] define similarity metrics based on the assumption that the length of trajectories is the same. However, it does not make sense for some scenarios where the length of trajectories varies because of sporadic sampling. Moreover, LCSS [18] and CATS [21] use manually defined thresholds to match positions in the two trajectories. SST [32] is proposed in the work to measure trajectory similarity based on the spatial-temporal distance of matching point pairs across trajectories. Their performance heavily relies on the parameter settings, which are difficult to determine and are not flexible for sporadic and heterogeneous sampling.

There has been some works focusing on trajectory linking, which is one of the important applications of spatial-temporal similarity measure. FTL [1] merges two trajectories and defines the compatibility of a mutual segment based on a predefined threshold for velocity. In FTL, a global velocity threshold is used for all objects. Based on a similar concept of FTL, ST-Link [22] and SLIM [23] restrict the matching events to be within a window of a time units of each other, and use a manually predefined maximum speed to determine whether two objects are likely to have co-locations. Compared with the above works, the method we proposed (STS) does not require preknowledge of object speed. STS uses a personalized speed model to extract the speed distribution for any individual object and estimate its transition probability, which is more reasonable to consider the object mobility. Furthermore, DPLink [2] proposes an end-to-end deep learning based framework to link trajectories of the same users from different data sources. It relies on a large amount of training data to learn a feature extractor for extracting representative features for trajectories.

To mitigate the impact of sporadic data sampling, some works estimate the locations of an object for better similarity measurement. EDwP [15] and STED [33] use linear interpolation to model user mobility based on the assumption that objects do not change their direction between two adjacent sampled locations, which is too strong for some scenarios. Furthermore, Markov model and Brownian Bridge are often used to estimate an objecct location in-between two observations in a trajectory. For example, APM [34] and some works (such as [24], [25], [34]) utilize the Markov model to estimate user location. Transition probability between two locations in these works are based on the frequency of transitions in historical data. However, the estimated probability in these works is universal for all users, and these approaches may suffer from the data sparsity problem and the over-fitting problem [35]. Instead, STS uses a personalized transition model to estimate the transition probability for any individual user given its trajectory without any need for historical data of other users. For a Brownian Bridge, motion is also assumed to be a Gaussian random walk, and the Brownian Bridge allows to estimate the location in-between two discrete observations [36], [37]. In STS, object motion is estimated based on its speed probability distribution. It can be any arbitrary distributions and without the assumption that the form of the probability distribution is known. Brownian Bridge can be seen as a special case of our estimation approach when the speed probability distribution is assumed to be a Gaussian distribution.

## III. Preliminary

In this section, we first define path and trajectory in Section III-A, and then overview the proposed Spatial-Temporal Similarity (STS) in Section III-B.

### A. Path and Trajectory

*Definition 1:* **(Path)** A path refers to the actual movement of an moving object, which can be defined as a continuous function $f : T \rightarrow L$ where $T$ refers to time space and $L$ is the geographical space.

*Definition 2:* **(Trajectory)** A trajectory $Tra$ is a sequence of locations, each associated with a timestamp, describ-

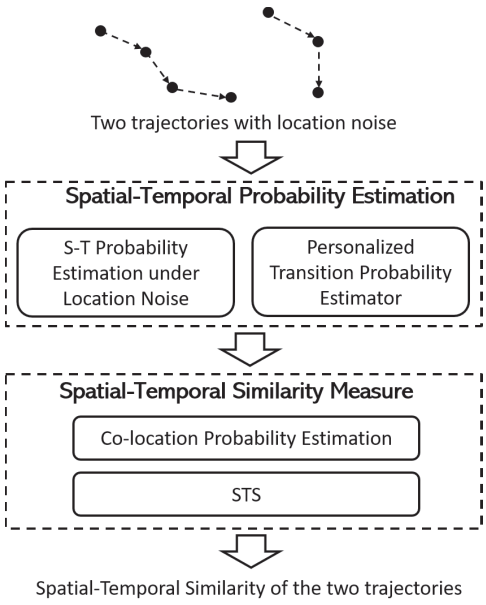



Fig. 2. An overview of STS.

ing the movement of an object. It is defined as $Tra = \{(\ell_1, t_1), (\ell_2, t_2), ..., (\ell_n, t_n)\}$, where $\ell_i$ is a location and $t_i$ is the associated timestamp.

A *trajectory* is a discrete representation of an object path, which can be viewed as a sampling process from this path. With various settings of sampling rates and observed duration, the length of trajectories is usually not the same.

### B. STS Overview

We overview the proposed STS in Figure 2. STS contains two important modules, the *spatial-temporal probability estimation* and the *spatial-temporal similarity measure*.

Given the location noise distribution of any location in a trajectory, STS uses the spatial-temporal probability estimation approach to estimate the probability distribution of the object's locations at any time (i.e., spatial-temporal probability). The estimation approach consists of two components: the *spatial-temporal (S-T) probability estimation under location noise* and the *transition probability estimation*.

- *Spatial-temporal (S-T) probability estimation under location noise:* By taking the existence of location noise in a trajectory into account, each location in a trajectory is modeled as an observable outcome from a probability distribution over some grids instead of a deterministic point, i.e., location probability distribution. An object's spatial-temporal probability distribution is then estimated from the location probability distributions in a trajectory and the transition probabilities between the location probability distributions.
- *Transition probability estimation:* To estimate an object's transition probability between locations in a time interval, we propose using the probability of the object's speed to denote its transition probability. To this end, we propose a kernel density estimation approach to estimate an object's personalized speed probability distribution given her/his trajectory. Based on that, an object's transition probability between any two locations in a time interval is then defined. Note that the estimated speed probability distribution is personalized for any individual user. Moreover, only the location data in an object's trajectory is used to estimate its speed probability distribution, and no training data from other objects is required.

Based on the estimated spatial-temporal probability, the spatial-temporal similarity measure is then formulated.

- *Co-location probability:* Given the spatial-temporal probability of objects, the probability of them being concurrently located at a grid at time $t$ can be estimated, even if the location is not observed in a trajectory. Their co-location probability at a timestamp $t$ can hence be derived as the sum of the co-location probability at all grids of the spatial space at $t$.
- *STS:* It is formulated as the average co-location probabilities at all timestamps in the two trajectories. For example, given two trajectories $Tra = \{(\ell_1, t_1), (\ell_2, t_2), (\ell_3, t_3)\}$ and $Tra' = \{(\ell'_1, t'_1), (\ell'_2, t'_2), (\ell'_n, t'_n)\}$, their spatial-temporal similarity is measured by their average co-location probabilities at times $\{t_1, t_2, t_3, t'_1, t'_2, t'_3\}$.

## IV. Spatial-Temporal Probability Estimation

In this section, we propose the spatial-temporal (S-T) probability estimation to estimate how likely an object is to be located at a grid at any time $t$ given her/his trajectory $Tra$. We first derive the S-T probability under location noise in Section IV-A. Then, we propose an approach to estimate an object's speed probability distribution given its trajectory. Based on that, we define the transition probability between locations in Section IV-B, which is an important component for S-T probability estimation.

### A. S-T Probability Estimation under Location Noise

We first partition the entire spatial area of interest (e.g., a city or a shopping mall) into $n$ disjoint but equal-sized grids, denoted as $R = \{r_1, r_2, \dots r_n\}$. Without loss of general, we use the central of grids to denote their locations. Based on these grids, we then estimate the probability $P(r_i, t|Tra)$ that an object is located at $r_i$ at $t$ given its trajectory $Tra$. According to the definition of conditional probability, the probability of an object being located at $r_i$ at time $t$ can be formulated as:

$$P(r_i, t|Tra) = \frac{P((\ell_1, t_1), ..., (\ell_i, t_i), (r_i, t), (\ell_{i+1}, t_{i+1}), ..., (\ell_n, t_n))}{P((\ell_1, t_1), ..., (\ell_i, t_i), (\ell_{i+1}, t_{i+1}), ...(\ell_n, t_n))}, \quad (1)$$

where $t_i \leq t \leq t_{i+1}$.

User transition between locations has been usually modelled with the Markov process in many prior works, such as [24], [25], [34], [35]. Formally, $P(\ell_i, t_i|\ell_{i-1}, t_{i-1}, ..., \ell_1, t_1) =$

$P(\ell_i, t_i|\ell_{i-1}, t_{i-1})$ for all $i = 2, 3, ..., n$. Therefore, we can simplify the conditional probability $P(r_i, t|Tra)$ into

$$P(r_i, t|Tra) = \frac{P(r_i, t|\ell_i, t_i)P(\ell_{i+1}, t_{i+1}|r_i, t)}{P(\ell_{i+1}, t_{i+1}|\ell_i, t_i)}, \quad (2)$$

where $P(r_i, t|\ell_i, t_i)$ denotes the transition probability that an object moves from $\ell_i$ at $t_i$ to $r_i$ at $t$, which will be discussed in Section IV-B.

Locations in a trajectory are usually noisy because of signal fluctuation or estimation error. In reality, location noise distribution of locations in a trajectory is available for some localization techniques, like GPS[1]. To consider the effect of location noise in S-T probability estimation, locations in a trajectory should be regarded as having a probabilistic distribution over the space instead of a deterministic spatial point.

Given the noise distribution $f$ at locations and an observed position $(\ell, t)$ in a trajectory $Tra$, the probability of the object being located at a grid $r$ at time $t$ is denoted as $f(r, \ell)$. When an object is observed to be at $\ell$ at $t$, $f(r, \ell)$ reflects the likelihood that the object's ground-truth location at $t$ is $r$.

Note that the given location noise distribution $f$ can be any arbitrary probability distribution. For ease of illustration, we use Gaussian distribution as a special case in this paper, as it has been widely used to model the location noise for localization systems [38]. Given the location noise $\sigma$ of the localization system and an observed position $(\ell_i, t_i)$ in a trajectory $Tra$, the probability of the user actually being located at a grid $r$ at time $t$ is

$$f(r, \ell_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{dis(\ell_i, r)}{2\sigma^2}\right), \quad (3)$$

where $dis(\ell_i, r)$ is the distance between $\ell_i$ and $r$.

A trajectory can then be further represented as a sequence of probability distributions: $\hat{Tra} = \{(D_1, t_1), (D_2, t_2), ..., (D_n, t_n)\}$, where $D_i = \{(r_j, f(r_j, \ell_i))|r_j \in R\}$ is the probability distribution over grids for the observed location $\ell_i$ in the trajectory $Tra$, and $r_j$ is any arbitrary grid in the space. Note that the location probability form is a generalized form for a trajectory since we can get the original trajectory if we set $D_i$ as $\ell_i$ with probability 1.

By considering the location noise, $(\ell_i, t_i)$ and $(\ell_{i+1}, t_{i+1})$ in Equation (2) should be modeled as probability distributions over the spatial space. Thus, the S-T probability estimation for $\hat{Tra}$ can be rewritten as

$$\approx \frac{P(r_i, t|\hat{Tra})}{\frac{\sum_{r_j \in R}(f(r_j, \ell_i) \cdot P(r_i, t|r_j, t_i)) \cdot \sum_{r_k \in R}(f(r_k, \ell_{i+1}) \cdot P(r_k, t_{i+1}|r_i, t))}{\sum_{r_j \in R}\sum_{r_k \in R} f(r_j, \ell_i) \cdot f(r_k, \ell_{i+1}) \cdot P(r_k, t_{i+1}|r_j, t_i)}}, \quad (4)$$

where $t_1 \leq t_i < t < t_{i+1} \leq t_n$, $R$ is a set of grids, $f(r_j, \ell_i)$ is the probability of the object being located at $r_j$ at $t_i$ given the observed position $(\ell_i, t_i)$ in its trajectory (Equation (3)), and $P(r_i, t|r_j, t_i)$ is the transition probability of moving from $r_j$ to $r_i$ in the time duration $|t_i - t|$.

Given an observed position $(\ell_i, t_i)$ in an object's trajectory, the object may be located at any grid $r_j$ with different probabilities in the spatial space. Thus, considering the effect of location noise, the transition probability term $P(r_i, t|\ell_i, t_i)$ in Equation 2 is replaced by the term $\sum_{r_j \in R}(f(r_j, \ell_i) \cdot P(r_i, t|r_j, t_i))$, which is the sum of probabilities over grids. The terms $P(\ell_{i+1}, t_{i+1}|r_i, t)$ and $P(\ell_{i+1}, t_{i+1}|\ell_i, t_i)$ in Equation 2 are also rewritten in Equation 4 to consider the location noise based on the defined grids accordingly.

Above all, the S-T probability $STP(\ell, t, Tra)$ of an object being located at a grid $r_j$ at time $t$ given a trajectory $Tra$ can be denoted as

$$STP(r_j, t, Tra) = \begin{cases} f(r_j, \ell_i), & \exists\ t_i = t, \\ P(r_j, t|\hat{Tra}), & t_1 \leq t_i < t < t_{i+1} \leq t_n, \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

If a position $(\ell_i, t_i)$ is observed in the trajectory, we use $f(r_j, \ell_i)$ to calculate the S-T probability (Equation (3)); if no position is observed at $t$ but $t_1 < t < t_n$, we use $P(r_j, t|\hat{Tra})$ for the computation (Equation (4)); and 0 otherwise.

### *B. Transition Probability Estimation*

In our S-T probability estimation, the transition probability $P(\ell^{'}, t^{'}|\ell, t)$ refers to the probability of an object moving from $\ell$ to $\ell^{'}$ in a time interval $|t - t^{'}|$. Hence, we propose using the probability of the object's speed of moving from $\ell$ to $\ell^{'}$ to denote the transition probability, which integrates both the spatial and temporal information in an object's mobility.

As observed in a prior work [26], the probability distribution of speed is distinct for different users, which is relevant to many factors such as gender, age and scenario. Therefore, instead of using a universal speed distribution for all users, we propose to use kernel density estimation to model a speed distribution for any individual object given its trajectory, i.e., each trajectory will have its personalized speed distribution. Kernel density estimation is a non-parametric way to estimate the probability density function of a random variable. It can be used with arbitrary distributions and without the assumption that the form of the probability distribution is known [39].

The kernel density estimation consists of two steps: speed sample collection and probability density estimation. To estimate the speed probability density of a trajectory, we firstly compute the speed between any two consecutive positions in the trajectory. Let $S$ be the set of speed samples drawn from some distribution with an unknown density $Q$. Its kernel density estimator over a speed $v$ using $S$ is given as

$$\hat{Q}(v) = \frac{1}{h|S|} \sum_{v^{'} \in S} K\left(\frac{v - v^{'}}{h}\right), \quad (6)$$

where $|S|$ denotes the number of samples, $K(\cdot)$ is the kernel (a non-negative function), and $h > 0$ is a smoothing parameter called the bandwidth. We exploit the most popular normal

[1]https://developer.mozilla.org/docs/Web/API/Geolocation/

kernel and the optimal bandwidth [40]: $h = \left(\frac{4\hat{\sigma}^5}{3|S|}\right)^{1/5}$, where $\hat{\sigma}$ is the standard deviation of the samples. Therefore, the transition probability of moving from a location $\ell$ at $t$ to another location $\ell^{'}$ at $t^{'}$ is computed as:

$$\begin{aligned} P(\ell^{'}, t^{'}|\ell, t) &= h \cdot \hat{Q}\left(\frac{dis(\ell, \ell^{'})}{|t - t^{'}|}\right) \\ &= \frac{1}{|S|}\sum_{v^{'} \in S} K\left(\frac{v - v^{'}}{h}\right), \end{aligned} \tag{7}$$

where $dis(\ell, \ell^{'})$ is the distance between $\ell$ and $\ell^{'}$, and $v = dis(\ell, \ell^{'})/(|t - t'|)$. It is worth noting that we only use the location data in an object's trajectory to estimate its speed probability distribution, and no other historical data are needed in our approach.

## V. STS: Spatial-Temporal Similarity Measure

In this section, based on the estimated S-T probability of objects, we propose the co-location probability estimation approach in Section V-A. After that, we take the average of the co-location probabilities at timestamps in the two trajectories to denote their spatial-temporal similarity, which is presented in Section V-B. Finally, we provide the computation complexity analytics in Section V-C.

### A. Co-location Probability Estimation

With the S-T probability estimation, the co-location probability of two trajectories at any time $t$ can be estimated, even if the location at $t$ is not observed in a trajectory. Given trajectories $Tra_1$ and $Tra_2$ of two objects, their co-location probability $CP(r, t|Tra_1, Tra_2)$ at a grid $r$ at time $t$ is defined as

$$CP(r, t|Tra_1, Tra_2) = STP(r, t, Tra_1) \cdot STP(r, t, Tra_2), \tag{8}$$

where $STP(r, t, Tra_1)$ and $STP(r, t, Tra_2)$ is the probability of two objects being located at $r$ at $t$ given their trajectories, respectively (Equation (5)).

Consequently, the co-location probability of $Tra_1$ and $Tra_2$ at a time $t$ can be approximated as:

$$\begin{aligned} &CP(t|Tra_1, Tra_2) \\ \approx &\sum_{r \in R} CP(r, t|Tra_1, Tra_2) \\ = &\sum_{r \in R} \left(STP(r, t, Tra_1) \cdot STP(r, t, Tra_2)\right), \end{aligned} \tag{9}$$

where $R$ is a set of grids.

Given two trajectories $Tra$ and $Tra'$ with their noise distribution $f(\cdot)$ and $f'(\cdot)$, and a set of grids $R$, the computation of the co-location probability of two trajectories at a time $t_i$ is presented in Algorithm 1. If the locations at $t_i$ are both observed in $Tra$ and $Tra'$ (Line 4), we compute the location probability at each grid for both trajectories using Equation 3, and normalize them (Line 5∼8). The co-location probability at $t_i$ is calculated as the sum of the co-location probability at all grids (Line 10). Otherwise, if the location at $t_i$ is observed in $Tra$ but not in $Tra'$ (Line 13), we compute the location probability at grids for $Tra$ using Equation 3, and for $Tra'$ using Equation 4, followed by the normalization (Line 14∼17). As the computation of the denominator in Equation 4 is the same for all grids at $t_i$, we do not have to calculate it due to the normalization. Based on these probabilities, the co-location probability can then be calculated (Line 18∼20). Similarly, if the location at $t_i$ is only observed in $Tra'$ but not in $Tra$, the computation process is shown in Lines 22 to 31.

**Algorithm 1:** Co-location probability of two trajectories at a time $t_i$.

1 **Input:** Two trajectories $Tra$, $Tra'$, their noise distribution $f(\cdot)$, $f'(\cdot)$, a time stamp $t_i$, and a set of grids $R$;
2 **Output:**The co-location probability $CP$ of $Tra$ and $Tra'$ at $t_i$ .
3 $CP = 0$;
4 **if** $t_i$ *in* $Tra$ *and* $t_i$ *in* $Tra'$ **then**
5  **foreach** $r$ *in* $R$ **do**
6  Compute $f(r, \ell_i)$ and $f'(r, \ell_i')$;
7  **end**
8  Normalize $f(r, \ell_i)$ and $f'(r, \ell_i')$;
9  **foreach** $r$ *in* $R$ **do**
10  $CP+ = f(\ell, \ell_i) \times f'(\ell, \ell_i')$;
11  **end**
12 **else**
13  **if** $t_i$ *in* $Tra$ **then**
14  **foreach** $r$ *in* $R$ **do**
15  Compute $f(r, \ell_i)$ and $P(r, t|\hat{T}ra^{'})$;
16  **end**
17  Normalize $f(r, \ell_i)$ and $P(r, t|\hat{T}ra^{'})$;
18  **foreach** $r$ *in* $R$ **do**
19  $CP+ = f(r, \ell_i) \times P(r, t|\hat{T}ra^{'})$;
20  **end**
21  **else**
22  **foreach** $r$ *in* $R$ **do**
23  Compute $f'(r, \ell_i)$ and $P(r, t|\hat{T}ra)$;
24  **end**
25  Normalize $f'(r, \ell_i)$ and $P(r, t|\hat{T}ra)$;
26  **foreach** $r$ *in* $R$ **do**
27  $CP+ = P(r, t|\hat{T}ra) \times f'(r, \ell_i')$;
28  **end**
29  **end**
30 **end**
31 return $CP$;

### B. Spatial-Temporal Similarity

The spatial-temporal similarity STS of two trajectories is defined as the average of co-location probabilities at all

timestamps in the two trajectories:

$$STS(Tra, Tra') = \frac{\sum_{i=1}^{|Tra|} CP(t_i|Tra, Tra') + \sum_{j=1}^{|Tra'|} CP(t_j|Tra, Tra')}{|Tra| + |Tra'|}, \quad (10)$$

where $|Tra|$ and $|Tra'|$ are the length of the two trajectories respectively, and $CP(t_i|Tra, Tra')$ is the two trajectories' co-location probability at $t_i$ (Equation 9).

As the length of different trajectories varies, the number of co-location probabilities to be evaluated differs for different trajectory pairs. To alleviate the impact of the length of different trajectories, we use the average of co-location probabilities as their spatial-temporal similarity.

### C. Computation Complexity

We first discuss the complexity of computing the transition probability (Equation 7), following the discussion on the Algorithm 1 for Equation 8. Finally, we present the total time for the computation of spatial-temporal similarity (Equation 10).

To calculate the transition probability (Equation 7), we first traverse locations in a trajectory for speed sample collection, the time complexity of which is $O(|Tra|)$. Once the speed sample is obtained, it takes $O(|S|)$ to compute the transition probability, where $|S| = |Tra| - 1$ is the size of the speed samples.

As shown in Algorithm 1, there are three possible cases for computing the co-location probability of two trajectories at a time. The time complexity for the first case (Line 4 $\sim$ 11) is $O(|R|)$. In the second case (Line 13 $\sim$ 20), the computation and nomalization of $f(r, \ell_i)$ take $O(|R|)$ for all $r$ in $R$. As the computation of the denominator in Equation 4 is the same for all grids at $t_i$, we do not have to calculate it due to the normalization. The computation complexity is hence $O(|R|^2 \times |Tra'|)$. Thus, the time complexity of the second case is $O(|R|^2 \times |Tra'| + |R|)$. Similarly, it takes $O(|R|^2 \times |Tra| + |R|)$ for the computation for the third case (Line 22 $\sim$ 28).

Thus, the worst cast of total time complexity for computing $STS$ using Equation 10 is $O(|Tra| \times (|R|^2 \times |Tra'| + |R|) + |Tra^{'}| \times (|R|^2 \times |Tra| + |R|)) = O(|Tra| \times |Tra^{'}| \times |R|^2)$.

## VI. Illustrative Experimental Results

We have conducted extensive experiments on two real datasets to evaluate the performance of STS. In this section, we first introduce the datasets and baselines in Section VI-A, followed by the performance metrics (Section VI-B). Then, we compare the performance of STS with the state-of-the-art approaches on the task of trajectory matching for different data sampling rates, heterogeneous data sample rates, location noise and different components in Section VI-C. Furthermore, we compare their performance on cross-similarity deviation with respect to heterogeneous data sampling in Section VI-D. Finally, we discuss the effect of grid sizes on STS in Section VI-E.

### A. Datasets and Baselines

We evaluate the performance of STS using trajectory data collected outdoors and indoors. The description of the two datasets used in our experiments is as follows.

- *Taxi dataset:* The taxi dataset[2] $D_T$ was collected by all the 422 taxis running in the city of Porto, in Portugal over 12 months. These taxis operate through a taxi dispatch center, using mobile data terminals installed in the vehicles to collect the location data. Each taxi reports its location every 15 seconds. The trajectory dataset contains 1.7 million trajectories. In our experiments, we removed trajectories the length of which was less than 20 so that we could sample sub-trajectories with different sampling rates to evaluate the effect of low and heterogeneous data sampling rates.
- *Shopping mall dataset:* The shopping mall dataset $D_S$ was collected by pedestrians in a large shopping mall. We deployed a WiFi fingerprint-based sensing system in a large shopping mall to collect pedestrians' location data [41]. The shopping mall consists of stores, corridors and some open space. Locations of pedestrians whose mobile devices have WiFi on would be collected by our sensing system. A record in the system consists of the device's MAC address, the coordinate of the device's location, and the timestamp. To construct trajectories, we group the location data based on the MAC address, and sort them by the timestamp. In our experiment, we collected $896,900$ records of location data of $12,858$ MAC addresses from 08:00 to 22:00 in one day, forming $12,858$ trajectories. For the purpose of the experiments, we removed trajectories the length of which was less than 20, which yielded $1,561$ trajectories.

We compared our proposed model with the following state-of-the-art models:

- CATS [21]: The Clue-Aware Trajectory Similarity (CATS) is a metric for measuring trajectory similarity, which aims to couple as many spatially and temporally co-located data points between two trajectories. CATS relies on two manually defined parameters to tackle the challenges of location noise and heterogeneous data sampling.
- EDwP [15]: Edit Distance with Projections (EDwP) is a robust distance function to quantify the similarity between trajectories. It has been proved to be efficient for similarity measurement under condition of inconsistent and variable sampling rates. It uses the linear interpolation to infer a user's location to address the issue of sporadic and heterogeneous sampling.
- APM [34]: APM uses a trajectory calibration process to transform a heterogeneous trajectory dataset to one with unified sampling strategies. In our experiments, we divide the space into grids, and use the centrals of grids as the anchor points for calibration. DTW [13] is used as the similarity metric after calibration.

[2]http://www.geolink.pt/ecmlpkdd2015-challenge/dataset.html

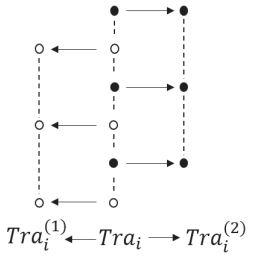



Fig. 3. Sample two sub-trajectories $Tra_i^{(1)}$ and $Tra_i^{(2)}$ from a raw trajectory $Tra_i$.

- KF: Kalman filter (KF) is an algorithm to estimate unknown variables that tend to be more accurate than those based on a single measurement. It is used to estimate the object location at a given time in our experiments. After the locations are estimated, we use DTW [13] for similarity comparison.
- WGM [19]: WGM measures similarity as the arithmetic mean of point-wise distances (e.g., origin vs. origin and destination vs. destination), each achieved through the weighted geometric mean of Euclidean similarity (spatial) and their temporal similarity.
- SST [32]: SST measures the similarity by synchronously matching the spatial distance against temporal distance. It matches points of two trajectories using the strategy of minimal point-to-segment similarity and maximal point-to-point similarity.

We do not include traditional similarity metrics such as DTW, LCSS, EDR in our experiments, since CATS and EDwP have been proved to outperform them in many previous works [15], [16], [21].

We implement STS, CATS, APM, KF, WGM and SST using Python. EDwP is implemented by the authors of the work [15] using Java, which is available online. The default grid sizes are set as 100m $\times$100m and 3m $\times$3m for the taxi dataset and the shopping mall dataset, respectively. The experiment settings for baseline approaches are adopted as introduced in prior works.

### B. Performance Metrics

One of the most important applications for spatial-temporal similarity measurement is trajectory matching [1], [16]. In a space with various types of sensing systems, a user leaves multiple trajectories for different sensing systems. Given two sets of trajectories $D^{(1)}$ and $D^{(2)}$ collected by different sensing systems, an effective similarity measure should match correctly two trajectories of the same user, namely identifying a user's trajectory in $D^{(1)}$ as the most similar one to her/his trajectory in $D^{(2)}$. Thus, we evaluate the performance of STS and other baseline approaches on trajectory matching. The experiment design is similar to that of prior works [15], [16].

Assume that $Tra_i^{(1)} \in D^{(1)}$ and $Tra_i^{(2)} \in D^{(2)}$ are trajectories from the same objects. For each trajectory $Tra_i^{(1)}$ in $D^{(1)}$, we measure the similarity of $Tra_i^{(1)}$ and any trajectories from $D^{(2)}$. We sort the trajectories in $D^{(2)}$ with respect to the similarity, and denote the rank of $Tra_i^{(2)}$ as $r_i$. Based on that, two performance metrics, precision and mean rank, which have been used to evaluate the performance of trajectory matching in prior works, are used for evaluation.

- Precision: If $r_i = 1$, we define $p_i$ for $T_i^{(1)}$ as 1, and 0 otherwise. Thus, the precision $P$ is defined as

$$P = \frac{(\sum_{i=1}^{n} p_i)}{n}. \tag{11}$$

- Mean rank: It is defined as the average of all $r_i$:

$$MR = \frac{(\sum_{i=1}^{n} r_i)}{n}. \tag{12}$$

The performance of trajectory matching in terms of precision and mean rank will be discussed in Section VI-C.

Furthermore, a good similarity measure should be able to preserve the similarity between two different trajectories, regardless of the data sampling strategy. Thus, we use the metric cross-similarity deviation for evaluation, which is also used in the previous works [16] and [34]. The cross-similarity deviation is defined as follows:

$$\frac{|d(Tra_1, Tra_2^{'}) - d(Tra_1 - Tra_2)|}{|d(Tra_1 - Tra_2)|}, \tag{13}$$

where $Tra_1$ and $Tra_2$ are two distinct trajectories, and $Tra_2^{'}$ is a sub-trajectory of $Tra_2$ which is sampled from $Tra_2$ with a given sampling rate. A smaller cross-similarity deviation indicates that the measured similarity is closer to the ground-truth [16]. The comparison of cross-similarity deviation will be presented in Section VI-D.

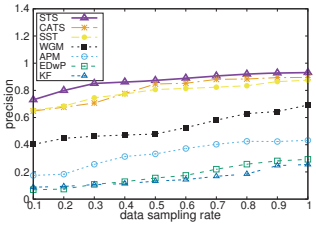
### C. Performance of Trajectory Matching

We compare STS with other state-of-the-art methods on the task of trajectory matching. We first introduce the construction of datasets, and then discuss the effect of low data sampling rates, heterogeneous data sampling rates, and location noise. Furthermore, we also evaluate the effectiveness of each component in STS on the trajectory matching.

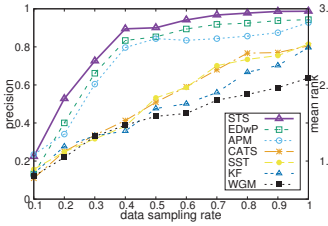
**Dataset construction** To overcome the lack of ground-truth, we construct the dataset following the prior work [16]. As shown in Figure 3, for each trajectory $Tra_i$ in a dataset, we sample two sub-trajectories by alternately taking points from it, denoted as $Tra_i^{(1)}$ and $Tra_i^{(2)}$, and use them to construct two new datasets $D^{(1)} = \{Tra_i^{(1)} | i = 1, 2, \ldots, n\}$ and $D^{(2)} = \{Tra_i^{(2)} | i = 1, 2, \ldots, n\}$. In the constructed dataset, $Tra_i^{(1)} \in D^{(1)}$ and $Tra_i^{(2)} \in D^{(2)}$ belong to the same object.

We perform the construction approach on the taxi dataset $D_T$ and the bike dataset $D_S$ respectively, and obtain two pairs of new datasets $(D_T^{(1)}, D_T^{(2)})$ and $(D_S^{(1)}, D_S^{(2)})$. After that, we evaluate the performance of trajectory matching on these two pairs of new datasets, respectively.

**Effect of different data sampling rates:** The similarity of trajectories with a low data sampling rate will be challenging to measure. To study the effect of different data sampling

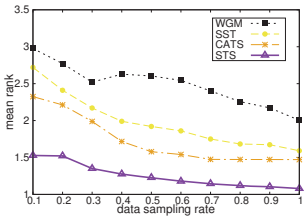



(a) Shopping mall dataset.

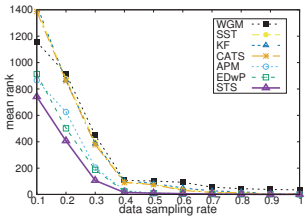



(b) Taxi dataset.

Fig. 4. Precision versus low data sampling rates.




(a) Shopping mall dataset.

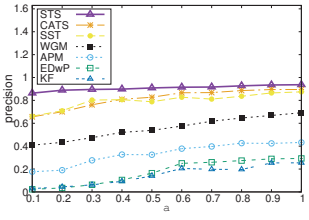



(b) Taxi dataset.

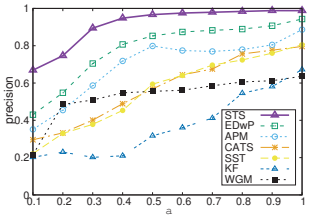
Fig. 5. Mean rank versus low data sampling rates.

rates, for each trajectory in $D^{(1)}$ and $D^{(2)}$, we sample a sub-trajectory with a sampling rate, which is set to be $0.1 \sim 0.9$.

The precision versus different data sampling rates is shown in Figure 4(a) (Shopping mall dataset) and Figure 4(b) (Taxi dataset). From Figure 4(a), we learn that as the data sampling rate increases, the precision of all approaches increases, because the location data become more dense in the trajectories. Compared with the state-of-the-art methods, STS has the highest precision for all data sampling rates. The difference in the precision of STS and other approaches becomes larger when the data sampling rate drops. STS makes a significant improvement for trajectories with low data sampling rates (e.g., around $12\%$ for CATS and SST, and $38\%$ for WGM when the data sampling rate is $0.1$ in our experiments), which demonstrates the effectiveness of STS to tackle the challenge of low data sampling rates. The result of the taxi dataset in Figure 4(b) can lead to some similar findings. We present the result of mean rank versus data sampling rate in Figure 5(a) (Shopping mall dataset) and Figure 5(b) (Taxi dataset). Because the mean rank of EDwP, APM and KF is too large

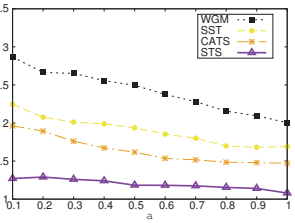



(a) Shopping mall dataset.

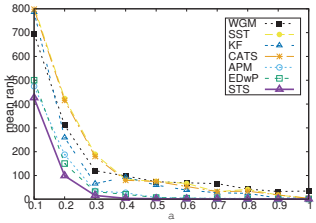



(b) Taxi dataset.

Fig. 6. Precision versus heterogeneous data sampling rates.




(a) Shopping mall dataset.



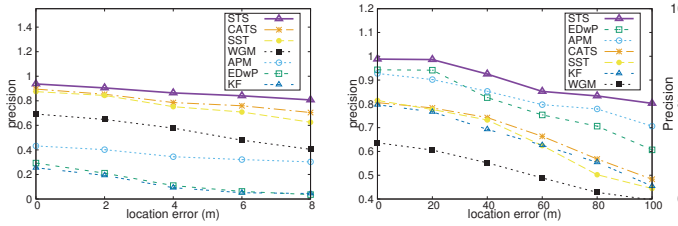

(b) Taxi dataset.

Fig. 7. Mean rank versus heterogeneous data sampling rates.

on the shopping mall dataset (from $9.28$ to $90.75$), we did not plot it in Figure 5(a). As the data sampling rate increases in Figure 5(a), the mean rank of all approaches decreases, indicating that the performance is improved. STS always outperforms other approaches, and the difference becomes more significant when the data sampling rate becomes lower. Similar results can be found from the mean rank of the taxi dataset (Figure 5(b)). However, compared with the shopping mall dataset, we find that precision is much lower and mean rank is much larger for the taxi dataset when the data sampling rates are low. The potential reason could be that there are more trajectories in the taxi dataset. Moreover, when the data sampling rate is extremely low ($90\%$ of data are filtered), some of the trajectories become very sparse, i.e., only a few locations in a trajectory. Consequently, the mean rank of these trajectories in taxi dataset may become extremely high. Meanwhile, EDwP has much better performance on the taxi dataset than on the shopping mall dataset, which reveals the limitation of EDwP in the indoor scenario due to its strong assumptions of user mobility. The performance of APM and KF also degrades significantly on the shopping mall dataset. The reason could be that the impact of location noise and sporadic data sampling becomes more severe in a narrow site, and the performance of the frequency-based transition estimation degrades significantly due to the more complex topological structures in a shopping mall (e.g., walls, stairs, etc.) Compared with other approaches, STS is more general and robust in different scenarios.

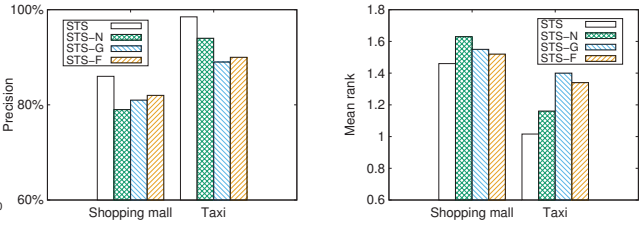
**Effect of heterogeneous sampling rates:** To evaluate the effect of sporadic sampling on the similarity measure, we discuss the precision and mean rank versus heterogeneous data sampling rates. For each trajectory in $D^{(2)}$, we sample a sub-trajectory with a sampling rate $\alpha$ and compute the similarity between the sub-trajectories and trajectories in $D^{(1)}$. A smaller $\alpha$ indicates a larger difference between two trajectories in the sampling rate. $\alpha$ is set as $\{0.1, 0.2, \ldots 0.9\}$ in the discussion.

Results of precision versus heterogeneous data sampling rates of the shopping mall dataset are shown in Figure 6(a). As the difference in sampling rate increases (the $\alpha$ decreases), the precision of all approaches decreases. The precision of STS is always higher than that of others. The improvement of STS becomes more obvious when the difference in the sampling rate increases (with an improvement of $20\%$ on CATS and $40\%$ on WGM). The result of mean rank is shown in Figure



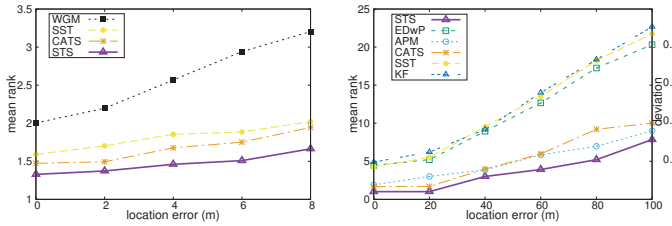

(a) Shopping mall dataset. (b) Taxi dataset.

Fig. 8. Precision versus location noise.



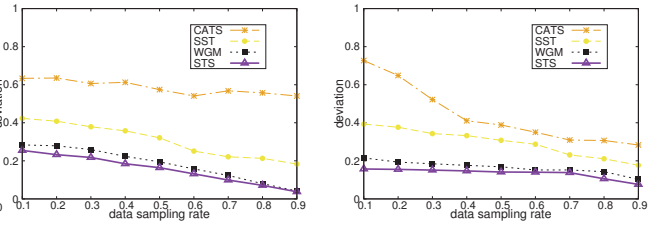

(a) Precision. (b) Mean rank.

Fig. 10. Performance of different variants.




(a) Shopping mall dataset. (b) Taxi dataset.

Fig. 9. Mean rank versus location noise.




(a) Shopping mall dataset. (b) Taxi dataset.

Fig. 11. Cross-similarity deviation of heterogeneous data sampling rates.

7(a), from which we can draw the consistent conclusion to the result of precision. We do not include the result of EDwP, APM and KF since its values are too large to plot in the figure. The results reveal that STS is effective to measure spatial-temporal similarity for trajectories with heterogeneous data sampling rates. A similar trend of the change can be observed in the results on the taxi dataset, which are presented in Figures 6(b) and 7(b).

**Effect of location noise:** To study the effect of location noise, we distort the location in trajectories from the datasets $D^{(1)}$ and $D^{(2)}$ by adding a Gaussian noise with radius $\beta$ meters as follows

$$\begin{aligned} x_i &= x_i + \beta \cdot d_x, d_x \sim \text{Gaussian}(0,1), \\ y_i &= y_i + \beta \cdot d_y, d_y \sim \text{Gaussian}(0,1). \end{aligned} \tag{14}$$

In our experiments, $\beta$ is set to be $[2m, 4m, 6m, 8m]$ for the shopping mall dataset, and $[20m, 40m, 60m, 80m, 100m]$ for the taxi dataset. Precision and mean rank are used as evaluation metrics.

Effect of location noise on precision and mean rank of the shopping mall dataset is presented in Figures 8(a) and 9(a), respectively. We take away the mean rank of EDwP, APM and KF since it is too large to plot in Figure 9(a). As the location noise increases, the precision of all approaches declines while the mean rank increases, indicating that the performance of all approaches declines when location noise becomes more severe. However, STS performs better than other approaches for different levels of location noise. Moreover, the performance difference of our approach and other baselines becomes more significant when the location noise becomes larger. Precision and mean rank versus location noise on the taxi dataset are presented in Figures 8(b) and 9(b), respectively. The result of WGM is taken away from Figure 9(b) as its value is too large. Our proposed metric also outperforms other baseline approaches on the taxi dataset. The results on the two datasets illustrates that it is more robust against location noise than others.

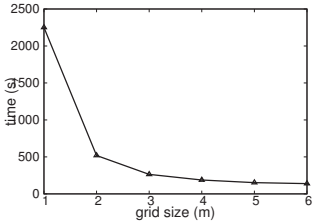
**Effectiveness of each component:** We evaluate the effectiveness of different components in STS by comparing STS with the following variants:

- STS-N: It does not consider location noise. Each location in STS-N is regarded as a deterministic spatial point instead of a probability distribution.
- STS-G: It does not consider personalized transition probability. Instead, it uses a constant global speed distribut ion for all objects.
- STS-F: It uses a frequency-based approach to estimate the transition probability between grids for all objects, which is also used in prior work, such as [24], [25], [34].

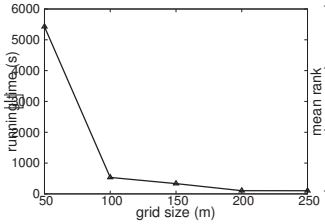
Following the previous experiment, we also distort the location in trajectories from the datasets $D^{(1)}$ and $D^{(2)}$. The location noise is set as 6m and 20m for the shopping mall dataset and the taxi dataset, respectively. The precision and mean rank on the two datasets are presented in Figures 10(a) and 10(b). STS outperforms STS-N on both datasets, indicating that the STS is effective in terms of considering location noise. Moreover, STS achieves higher precision and lower mean rank than STS-G and STS-F, which illustrates the effectiveness of our proposal personalized transition probability estimation approach.

### *D. Comparison of Cross-Similarity Deviation*

We further evaluate the performance of STS in terms of cross-similarity deviation. Since the performance of EDwP,

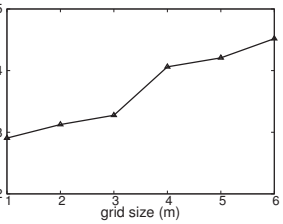



(a) Shopping mall dataset.

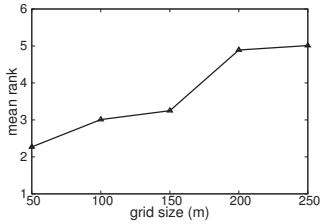



(b) Taxi dataset.

Fig. 12. Impact of grid sizes on efficiency.

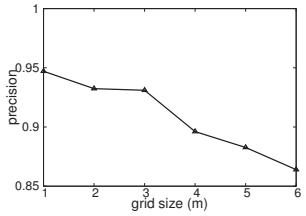



(a) Shopping mall dataset.

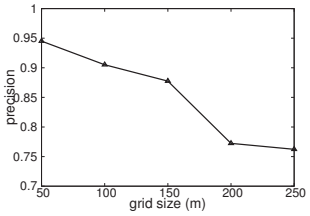



(b) Taxi dataset.

Fig. 14. Impact of grid sizes on mean rank.




(a) Shopping mall dataset.




(b) Taxi dataset.

Fig. 13. Impact of grid sizes on precision.

APM and KF is poor in the above evaluation, we only compare STS with CATS, WGM and SST in the following discussion. In our experiment, we randomly selected $1000$ pair of trajectories $(Tra_1, Tra_2)$ from a dataset. For each $Tra_2$, we down-sampled $9$ sub-trajectories from it with a different sampling rate $\alpha$, where $\alpha$ is set to be from $0.1$ to $0.9$.

The average of cross-similarity deviation (Equation 13) for different sampling rates is presented in Figure 11(a) (Shopping mall dataset) and Figure 11(b) (Taxi dataset). From the result of the shopping mall dataset, as the data sampling rate becomes larger, the cross distance deviation becomes smaller. That is because a larger sampling rate indicates a smaller difference between $Tra_2$ and $Tra_2^{'}$, and the difference between $d(Tra_1, Tra_2)$ and $d(Tra_1, Tra_2^{'})$ should be smaller. The cross-similarity deviation of STS is always smaller than that of other approaches. Although CATS has a good performance on the metric precision and mean rank, its performance is not as good as other approaches on the metric of cross-similarity deviation. A comparison of the results indicates that STS is able to preserve the distance between two different trajectories, regardless of the data sampling strategy. Consistent experiment results could be found on the taxi dataset, which is shown in Figure 11(b).

### *E. Grid Size*

A small grid size means a larger number of grids, leading to a better probability approximation but higher time cost. We discuss the effect of grid size on effectiveness and efficiency for STS. The precision and mean rank are used as metrics for effectiveness evaluation, and the running rime is used to evaluate the efficiency.

The results on the shopping mall dataset are presented in Figures 12(a), 13(a), 14(a). As the grid size increases, the precision and the running time decline while the mean rank increases on both datasets, which is consistent with our intuition. The decline in running time is not obvious for the shopping mall dataset when the grid size is larger than $3m$ (Figure 12(a)). Meanwhile, precision drops and mean rank increases dramatically when the grid size is larger than $3m$. As the location error of the localization technique used to collect the shopping mall dataset is also around $3m$, we suggest that the grid size could be set to be the same as the location error to balance the trade-off of the effectiveness and the efficiency. As for the taxi dataset, a grid size of $100$m $\sim$ $150$m could be a good choice considering the effectiveness and the efficiency.

## VII. Conclusion

We propose STS, a novel and effective measure to evaluate the spatial-temporal similarity between any pair of trajectories with location noise and sporadic location sampling. STS employs a spatial-temporal probability estimation approach to compute the probability distribution of the object location at any time. In the proposed estimation approach, each location in a trajectory is modeled as a probability distribution over space instead of a spatial point. Then the transition probability of an object between any two locations is estimated based on the personalized speed probability distribution drawn from the trajectory itself. Based on the estimated spatial-temporal probability of objects, their co-location probability can be estimated. Finally, STS used the average co-location probabilities of two trajectories to denote their spatial-temporal similarity. We conducted extensive experiments using two real datasets for taxis and a shopping mall. The results show that STS is substantially more accurate and robust against location noise and sporadic data sampling than the state-of-the-art approaches, with improvements of $63\%$ on precision and $85\%$ on mean rank. The excellent performance in the taxi and mall datasets illustrated that our proposed STS can be applied in both indoor and outdoor scenarios.

## Acknowledgment

This work was supported, in part, by Hong Kong General Research Fund under grant number 16200120.

## References

[1] H. Wu, M. Xue, J. Cao, P. Karras, W. S. Ng, and K. K. Koo, "Fuzzy trajectory linking," in *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*. Helsinki, Finland: IEEE, 2016, pp. 859–870.

[2] J. Feng, M. Zhang, H. Wang, Z. Yang, C. Zhang, Y. Li, and D. Jin, "Dplink: User identity linkage via deep neural network from heterogeneous mobility data," in *The World Wide Web Conference*. San Francisco: ACM, 2019, pp. 459–469.

[3] J. Cui, H. Zha, H. Zhao, and R. Shibasaki, "Multi-modal tracking of people using laser scanners and video camera," *Image and vision Computing*, vol. 26, no. 2, pp. 240–252, 2008.

[4] Q. Gao, F. Zhou, K. Zhang, G. Trajcevski, X. Luo, and F. Zhang, "Identifying human mobility via trajectory embeddings." in *IJCAI*, vol. 17. Melbourne, Australia: IJCAI, 2017, pp. 1689–1695.

[5] M. A. Andresen, A. S. Curman, and S. J. Linning, "The trajectories of crime at places: understanding the patterns of disaggregated crime types," *Journal of quantitative criminology*, vol. 33, no. 3, pp. 427–449, 2017.

[6] R. Sen, Y. Lee, K. Jayarajah, A. Misra, and R. K. Balan, "Grumon: Fast and accurate group monitoring for heterogeneous urban spaces," in *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*. Memphis, Tennessee: ACM, 2014, pp. 46–60.

[7] K. Jayarajah, Z. Lantra, and A. Misra, "Fusing wifi and video sensing for accurate group detection in indoor spaces," in *Proceedings of the 3rd International on Workshop on Physical Analytics*. Singapore: ACM, 2016, pp. 49–54.

[8] M.-H. Park, J.-H. Hong, and S.-B. Cho, "Location-based recommendation system using bayesian user's preference model in mobile devices," in *International conference on ubiquitous intelligence and computing*. Hong Kong, China: Springer, 2007, pp. 1130–1139.

[9] X. Tang, B. Gong, Y. Yu, H. Yao, Y. Li, H. Xie, and X. Wang, "Joint modeling of dense and incomplete trajectories for citywide traffic volume inference," in *The World Wide Web Conference*. San Francisco: ACM, 2019, pp. 1806–1817.

[10] Y. Zheng, "Trajectory data mining: an overview," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 6, no. 3, p. 29, 2015.

[11] H. Wu, W. Sun, B. Zheng, L. Yang, and W. Zhou, "Clsters: A general system for reducing errors of trajectories under challenging localization situations," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, p. 115, 2017.

[12] S. He and S.-H. G. Chan, "Tilejunction: Mitigating signal noise for fingerprint-based indoor localization," *IEEE Transactions on Mobile Computing*, vol. 15, no. 6, pp. 1554–1568, 2015.

[13] B.-K. Yi, H. Jagadish, and C. Faloutsos, "Efficient retrieval of similar time sequences under time warping," in *Proceedings 14th International Conference on Data Engineering*. Orlando, Florida, USA: IEEE, 1998, pp. 201–208.

[14] L. Chen, M. T. Özsu, and V. Oria, "Robust and fast similarity search for moving object trajectories," in *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*. Baltimore, Maryland: ACM, 2005, pp. 491–502.

[15] S. Ranu, P. Deepak, A. D. Telang, P. Deshpande, and S. Raghavan, "Indexing and matching trajectories under inconsistent sampling rates," in *2015 IEEE 31st International Conference on Data Engineering*. Seoul, South Korea: IEEE, 2015, pp. 999–1010.

[16] X. Li, K. Zhao, G. Cong, C. S. Jensen, and W. Wei, "Deep representation learning for trajectory similarity computation," in *2018 IEEE 34th International Conference on Data Engineering (ICDE)*. Paris: IEEE, 2018, pp. 617–628.

[17] Z. Chen, H. T. Shen, X. Zhou, Y. Zheng, and X. Xie, "Searching trajectories by locations: an efficiency study," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*. Indianapolis, Indiana, USA: ACM, 2010, pp. 255–266.

[18] M. Vlachos, G. Kollios, and D. Gunopulos, "Discovering similar multidimensional trajectories," in *Proceedings 18th international conference on data engineering*. San Jose, California: IEEE, 2002, pp. 673–684.

[19] R. Ketabi, B. Alipour, and A. Helmy, "Playing with matches: vehicular mobility through analysis of trip similarity and matching," in *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. Seattle, Washington, USA: ACM, 2018, pp. 544–547.

[20] E. Tiakas, A. N. Papadopoulos, A. Nanopoulos, Y. Manolopoulos, D. Stojanovic, and S. Djordjevic-Kajan, "Trajectory similarity search in spatial networks," in *2006 10th International Database Engineering and Applications Symposium (IDEAS'06)*. Delhi, India.: IEEE, 2006, pp. 185–192.

[21] C.-C. Hung, W.-C. Peng, and W.-C. Lee, "Clustering and aggregating clues of trajectories for mining trajectory patterns and routes," *The VLDB Journal—The International Journal on Very Large Data Bases*, vol. 24, no. 2, pp. 169–192, 2015.

[22] F. Basık, B. Gedik, Ç. Etemoğlu, and H. Ferhatosmanoğlu, "Spatio-temporal linkage over location-enhanced services," *IEEE Transactions on Mobile Computing*, vol. 17, no. 2, pp. 447–460, 2017.

[23] F. Basïk, H. Ferhatosmanoğlu, and B. Gedik, "Slim: Scalable linkage of mobility data," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2020, pp. 1181–1196.

[24] T. Emrich, H.-P. Kriegel, N. Mamoulis, M. Renz, and A. Zufle, "Querying uncertain spatio-temporal data," in *2012 IEEE 28th international conference on data engineering*. IEEE, 2012, pp. 354–365.

[25] J. Niedermayer, A. Züfle, T. Emrich, M. Renz, N. Mamoulis, L. Chen, and H.-P. Kriegel, "Probabilistic nearest neighbor queries on uncertain moving object trajectories," in *Proceedings of the VLDB Endowment*. Springer, 2014, pp. Vol. 7, No. 3.

[26] S. Chandra and A. K. Bharti, "Speed distribution curves for pedestrians during walking and crossing," *Procedia-Social and Behavioral Sciences*, vol. 104, pp. 660–667, 2013.

[27] H. Su, S. Liu, B. Zheng, X. Zhou, and K. Zheng, "A survey of trajectory distance measures and performance evaluation," *The VLDB Journal*, vol. 29, no. 1, pp. 3–32, 2020.

[28] L. Chen and R. Ng, "On the marriage of lp-norms and edit distance," in *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*. Hangzhou, China: VLDB Endowment, 2004, pp. 792–803.

[29] H. Wang, H. Su, K. Zheng, S. Sadiq, and X. Zhou, "An effectiveness study on trajectory similarity measures," in *Proceedings of the Twenty-Fourth Australasian Database Conference-Volume 137*. Adelaide, South Australia: Australian Computer Society, Inc., 2013, pp. 13–22.

[30] A. Ward, "A generalization of the frechet distance of two curves," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 40, no. 7, p. 598, 1954.

[31] N. Pelekis, I. Kopanakis, G. Marketos, I. Ntoutsi, G. Andrienko, and Y. Theodoridis, "Similarity search in trajectory databases," in *14th International Symposium on Temporal Representation and Reasoning (TIME'07)*. IEEE, 2007, pp. 129–140.

[32] P. Zhao, W. Rao, C. Zhang, G. Su, and Q. Zhang, "Sst: synchronized spatial-temporal trajectory similarity search," *GEOINFORMATICA*, 2020.

[33] M. Nanni and D. Pedreschi, "Time-focused clustering of trajectories of moving objects," *Journal of Intelligent Information Systems*, vol. 27, no. 3, pp. 267–289, 2006.

[34] H. Su, K. Zheng, H. Wang, J. Huang, and X. Zhou, "Calibrating trajectory data for similarity-based analysis," in *Proceedings of the 2013 ACM SIGMOD international conference on management of data*. New York,USA: ACM, 2013, pp. 833–844.

[35] H. Wu, J. Mao, W. Sun, B. Zheng, H. Zhang, Z. Chen, and W. Wang, "Probabilistic robust route recovery with spatio-temporal dynamics," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Francisco, California: ACM, 2016, pp. 1915–1924.

[36] T. W. Anderson and M. A. Stephens, "The continuous and discrete brownian bridges: Representations and applications," *Linear Algebra and its Applications*, vol. 264, pp. 145–171, 1997.

[37] J. S. Horne, E. O. Garton, S. M. Krone, and J. S. Lewis, "Analyzing animal movements using brownian bridges," *Ecology*, vol. 88, no. 9, pp. 2354–2363, 2007.

[38] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk, "On map-matching vehicle tracking data," in *Proceedings of the 31st international conference on Very large data bases*. Trondheim, Norway: VLDB Endowment, 2005, pp. 853–864.

[39] J.-D. Zhang and C.-Y. Chow, "igslr: personalized geo-social location recommendation: a kernel density estimation approach," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. Orlando, Florida, USA: ACM, 2013, pp. 334–343.

[40] B. W. Silverman, *Density estimation for statistics and data analysis*. British: Routledge, 2018.

[41] S. He and S.-H. G. Chan, "Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 466–490, 2015.