Spatial-temporal Mining for Urban Map-Matching

Josh Jia-Ching Ying, Bo-Nian Shi, Kun-Chan Lan and Vincent S. Tseng*
Institute of Computer Science and Information Engineering
National Cheng Kung University
No.1, University Road, Tainan City 701, Taiwan (R.O.C.)
{jashying, bernie0914}@gmail.com, klan@csie.ncku.edu.tw,
*Correspondence: tsengsm@mail.ncku.edu.tw

ABSTRACT
In recent years, the widespread of GPS devices produce huge amount of urban trajectories. Due to the complex structure of road network in urban area, many issues are not addressed in existing work, such as high volume of road network, diverse functionalities of roads, and high variance of road dynamics. In this paper, we propose a novel modularity-based map-matching algorithm called Urban Map-Matching (UrbMatch) utilizing urban GPS trajectories. UrbMatch considers (1) the spatial-temporal betweenness and (2) the marginal velocity of each road segment, to improve the accuracy of map-matching. Based on the results of spatial-temporal mining, a road network is decomposed into several sub-networks such that the map-matching task can be divided into several smaller sub-tasks and run in parallel. Accordingly, the running time can be reduced. We compare UrbMatch with an existing global map-matching algorithm using real-world dataset. The results show that our proposed approach not only are faster than state-of-the-art global map-matching method over 100 times but also outperforms other map matching techniques in terms of accuracy.

Categories and Subject Descriptors
H.2.8 [Database Management]: Database Applications – *Data Mining, Spatial Databases and GIS*

General Terms
Algorithms, Experimentation.

Keywords
Map Matching; Road Network Analysis; Trajectory Analysis; Centrality Mining

1. INTRODUCTION
With the rapid development of the global positioning system (GPS), driving GPS trajectories have become more and more accessible. These trajectories are useful for understanding users' moving behavior which is important for many applications such as location-based service, urban planning, and traffic management. However, due to the inherent errors of GPS data, it is essential to perform data cleansing on GPS trajectory traces before using it for any further analysis. The map matching technique, which matches all GPS points of trajectories onto road segments of a certain road network, has been widely used for such a data cleansing step.

To analyze the relationship between road segments and GPS trajectories, many existing map matching methods adopt the measurements of similarity between curves [1][2], geometric distance [3] [10], and similarity of vehicle velocity [14]. For example, Fréchet Distance [1][2] was originally proposed to measure how close two polygonal curves are. It is also able to measure the closeness of a GPS trajectory to a road segment. However, the above metrics are limited due to it's design for the matching of a single road segment at a certain point of time. Due to the complex structure of road network in the urban area, the dynamics of different road segments in a road network could affect each other [23]. For example, a serious traffic accident on a road segment could result in traffic jam in near-by road segments. Unfortunately, existing traditional map matching methods do not address the properties in the urban area including:

- **Diverse functionalities.** The functionality of different road segments might be various. For example, some road segments are used to bridge two areas (e.g. highways). Drivers commuting between these two areas would have a high probability of using these road segments. However, the traditional map matching methods do not consider the functionalities of different road segments and hence can not work well for the urban area.
- **High Volume.** The number of road segments in the urban area could be potentially large and therefore, given a GPS point, many possible road segments can be matched. However, the traditional map matching methods tend to perform the matching for the whole road network, which leads to significant running time when dealing with urban road networks.
- **Changing Velocity.** The dynamics of a road network in the urban area might changes significantly over time. For example, the marginal speeds of vehicles could change every hour. Nevertheless, traditional map matching methods generally either ignore the effect of vehicle velocitiesor using the same setting for a road segment over time (i.e., without considering temporal changes) for matching a trajectory onto the road network.

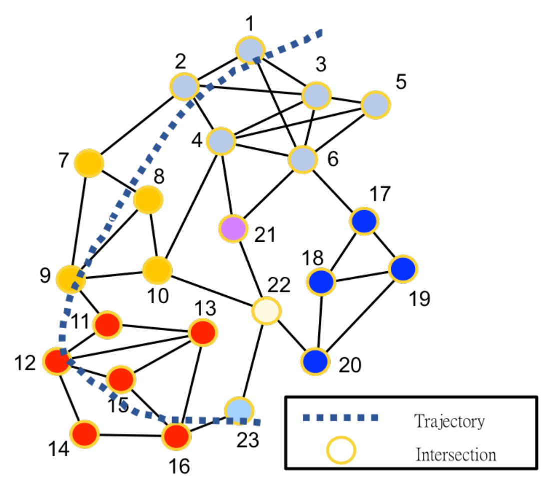


Figure 1. An Example of Road Network and GPS Trajectory

Intuitively, centrality analysis of road traffic dynamics could be useful for dealing with issues caused by "Diverse functionalities" of urban road networks [5] [8]. Take Figure 1 as an instance; the trajectory could be matched onto two possible road segments, <7, 9> and <8, 9>. If we focus only on the geometric distance, we should match the trajectory onto the road segment <8, 9>. However, according to the principle of centrality analysis, vehicles are more likely to pass along the road segment <7, 9> if they origin from road segment <1, 2> and destine to road segment <12, 15>. Although the centrality analysis has been discussed in many prior studies [5] [8], none of them considered both spatial and temporal factors of a road network.

Furthermore, to speed up the map-matching process, we proposed to divide whole road network into several smaller sub-network to parallel the map-matching tasks. For example, we can divide a road network into several sub-networks as shown in Figure 1, in which different sub-networks are represented by different colors. We can see that the road segment <9, 11> is only one way that connects the yellow and red sub-networks. Therefore, we can directly match the trajectory onto the road segment <9, 11> and parallel the matching task into two sub-tasks: matching the red sub-network and matching other sub-networks. Inherently, modularity optimization [15] provides a good solution for divide a network in to several sub-networks. The traditional modularity optimization, however, focuses only on the network structure, i.e., degree of vertex. In a real road network, many vehicles might not pass the vertexes with a high degree. Accordingly, the spatial-temporal centrality of road segments should be considered for modularity optimization for dividing a road network.

Finally, to understand dynamics of each road segment over time, the average speed of vehicles on the road can provide useful insights for dividing a road networks. With the advance of Geographic Information System (GIS), many digital map services can be accessed and utilized for route planning [6]. For example, the routes provided by Google Maps[1] are annotated with the traveling time which can be used to infer the current road speed for estimating the dynamics of a road network.

In this paper, we propose a novel map matching approach which is able to deal with issues caused by the complex structure of road networks in urban area. The core idea of our map-matching approach is to decompose a road network based on spatial-temporal information. To support our spatial-temporal based road network decomposition algorithm, we propose several metrics for representing both the spatial-temporal centrality and the real-time dynamic of each road segment. As mention earlier, the centrality of a road segment can be utilized for improving the accuracy of matching a GPS point onto the corresponding road segment. Owning to the lack of spatial-temporal information, traditional centrality can not be directly applied to measure the probability of matching a GPS point onto the road segment. Here we propose two kind of spatial-temporal centrality metrics, named *Spatial-temporal Betweenness* (*ST-Betweenness*) and *Spatial-temporal Betweenness Correlation* (*ST-Correlation*). Furthermore, a real-time vehicle speed estimator is proposed for evaluating the real-time dynamic of each road segment. The real-time vehicle speed estimator is developed based on the estimated time provided by Google Maps.

In summary, in this study we have a number of contributions as follows:

- We propose a novel two-phase map-matching method named *<u>Urb</u>an <u>Match</u>ing* (*UrbMatch*) for matching GPS trajectories onto a certain road network in the urban area.
- We define two new spatial-temporal metrics, called *ST-Betweenness* and *ST-Correlation*, to represent spatial-temporal relations among road segments in terms of betweenness centrality and closeness centrality. In addition, to consider real-time dynamics of road segments, we propose a metric based on Google Map API for measuring the marginal velocity of vehicles on each road segment across a day.
- We develop a spatial-temporal based road network decomposition algorithm, called *ST-Decomposition*, to decompose the whole road network based on the spatial-temporal information and marginal velocity.
- We use a map of Shanghai collected from Open Street map [25] and a real-world trajectory dataset obtained from WnSN, SJTU [28] to evaluate the performance of our proposal. The results show that our proposed approach not only are faster than state-of-the-art global map-matching method, ST-Matching [14], over 100 times but also outperforms other map matching techniques in terms of accuracy.

For the rest of this paper, we first briefly review the related work, and then describe our proposed Urban Map Matching system. Next, we present algorithms for spatial-temporal information mining, spatial-temporal based road network decomposition, and propose a two-phase map-matching models. Finally, we evaluate the performance of different components of our map-matching system and compare our approach with the existing map-matching method.

2. RELATED WORKS

In fact, map matching could be classified into three categories: a) local map matching [4] [7] [13] [19], b) global map matching [1] [2] [3] [14] [18] [20] [22], and c) multi-track map matching [9] [10] [11] [16] [17].

2.1 Local Map Matching

The local methods try to find partial match of geometries and combine all partial match as the result of map matching. To efficiently match trajectory onto a road network, Chawathe *et al.* proposes a segment-based matching method [4]. Meanwhile, a confidence measure is defined and assigned for different sampling points. To match the sampling points to the road segments, the sampling points with high confidence will be processed first, and then match trajectory segments with low confidence according to previously matched edges. The local map matching in [7] uses two closeness metrics to estimate the possibility of matching a GPS point onto a road segment. Meanwhile, one metric is to evaluate geographic distance, and the other is to evaluate orientation similarity. The combined value of metrics is computed as the total score for estimating the possibility of matching the GPS trajectory to the roads. Since all of GPS points of trajectory is examined for only one time, the time complexity is $O(n)$, where n is the number of GPS points of the trajectory. To deal with the complexity and inaccurate information, Liu *et al.* [13] propose a novel weighting method to describe likelihoods of candidate roads, using a distance forgetting factor for history accumulation. In [19], Wenk *et al.* proposed an adaptive clipping method by using Dijkstra algorithm to connect two partial matching results in shortest path manner. Accordingly, The time complexity of this

[1] https://support.google.com/maps/answer/2549020?hl=en

method is $O(mn \log m)$ (i.e., $O(n) \times O(m \log m)$), where m and n are number of edges in the road network and number of GPS points respectively. General speaking, a local map matching only considers a small portion of the trajectory to match road segments. It can deal with the simple road network structure. However, as the complexity of road network structure increasing in urban area, the candidate road segments might be crowded, causing significant damages of accuracy. By contrast, with reasonable scarcity of efficiency, our map-matching algorithm is more effective to the urban area.

2.2 Global Map Matching

The goal of global map matching is to match the entire trajectory with the road network. Most of existing works [1] [2] [3] are based on Fréchet distance which considers the continuity of curves. As the results, Fréchet distance is inherently suitable for measuring closeness between roads and trajectories. In [1] and [2], Alt *et al.* formula the map matching problem as a decision problem. First, all critical values could be detected by a parametric search algorithm. Then, the minimum Fréchet distance is determined by finding a monotone path in the free space from the lower left corner to the upper right corner. It runs in $O(mn \log^2 mn)$ time, where m and n are the number of edges and number of nodes in the road network respectively. Brakatsoulas *et al.* [3] extend Alt *et al.*'s work [1] by using average Fréchet distance to reduce the effect of outliers. Moreover, a variant Fréchet distance, called weak Fréchet distance, is proposed to speed-up Fréchet distance determine such that the complexity is reduced to $O(mn \log mn)$ time. In [20], the edit distance is adopted to measure the closeness between trajectory and matched road segments. Raymond *et al.* [18] proposed a simplification of the more-complex HMM-based method that maintains its capabilities to cope with the noises and sparsity of the raw GPS data. To deal with low sample rate problem, Lou *et al.* [14] propose several spatial-temporal metrics for improving the accuracy of map matching. Based on Lou *et al.*'s work [14], Yuan *et al.* proposed an interactive-voting based map-matching algorithm [22] to improve accuracy of the matching results. Global map matching aims either to search minimum Fréchet distance between the trajectory and the matched road segments or to formula map matching problem as optimization problem. However as the complexity of road network structure increasing in urban area, the missing of dead reckoning becomes more serious, causing significant damages of accuracy.

2.3 Multi-track Map Matching

The aim of multi-track map matching is to recover the regularity patterns among the input trajectories, and to preserve the regularity in the matched paths. Usually, multi-track map matching adopts data-driven techniques to improve the matching result using historical trajectories. In [17], Bayesian classifier that combines a Hidden Markov Model is utilized to model topological constraints of the road network for improving accuracy of map matching. Another general application [11] [16] is to utilize machine-learning techniques for training the parameters of the statistical models or the weight function. This approach makes it easier to tune the model, but it also has the tendency of over-fitting the predefined model. In [10], historical trajectories are directly processed into reference routes that can be queried to generate new routes. The algorithm is able to complete the matching even with very sparse data. However, as the generated path is solely based on a fairly accurate reference route database, it may fail at the case when a trajectory is sampled from a path not in the database.

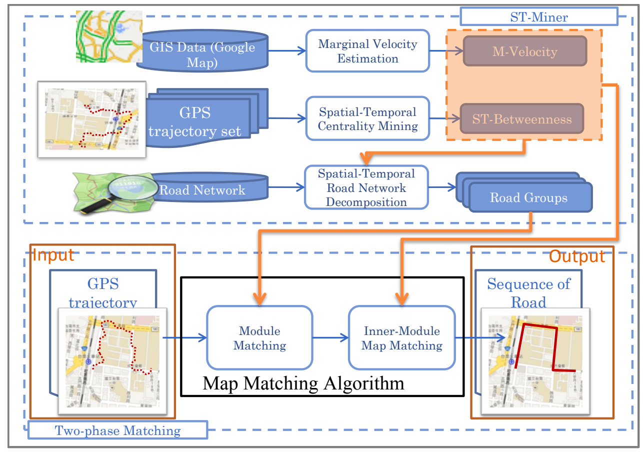


Figure 2. Framework of Urban Map Matching

3. OVERVIEW OF URBMATCH

With the notion of spatial-temporal information, we propose a novel Urban Map-matching framework, namely, *UrbMatch*, based on both the centrality and marginal velocity of road network. The proposed approach works for match a GPS trajectory onto an urban road network of which structure is very complex, e.g., Shanghai City. The *UrbMatch* framework consists of 1) an offline mining module and 2) an online matching module. Figure 2 shows the framework and its flow of data processing. The idea is to explore the spatial-temporal information of vehicles, captured in GPS trajectories, to improve accuracy of map-matching method. As shown, the mining module includes three models. The first model, called *Spatial-temporal Centrality Estimation* model, measures spatial-temporal betweenness of each road segment based on both the road network structure and the crawled GPS trajectories. The second model, called *Marginal Velocity Estimation* model, extracts each road segment's possible velocities from Google Maps. The third model, called *Road Network Decomposition* model, groups the road segments based on their dynamic. In the online module, we propose a two-phase method to evaluate the goodness for matching a trajectory onto a path in a road network. Here, we consider not only spatial-temporal information but also association information. First, we calculate the spatial-temporal score and derive several candidate paths. Then, the association score of each candidate path is evaluated. Finally, we compute a weighted average of geographic score and semantic score for each candidate path to select the most probable path for matching the trajectory over the road network.

4. ST-MINER

To our best knowledge, existing road network analysis metrics do not take into account the effect of spatial-temporal dynamic. Besides, all of graph theoretical metrics show fluctuate values in different districts (see section of Analysis of Shanghai Road Network). That means there are some potential effects related to spatial-temporal information. Therefore, it is necessary to provide some metrics which can reflect the effect of spatial-temporal information for improving accuracy of map-matching task. To compactly represent spatial-temporal information of each road segment, we propose three novel metrics, called *marginal velocity* (*M-Velocity*), *spatial-temporal betweenness* (*ST-Betweenness*) and *spatial-temporal betweenness correlation* (*ST-Correlation*). The M-Velocity can reflect the each road segment's velocity which is influenced by the dynamic of road segment. The

ST-Betweenness and the ST-Correlation can represent each road segment's spatial-temporal centrality and relation between each pair of road segments, respectively. Based on extracted spatial-temporal dynamic (i.e., M-Velocity, ST-Betweenness, and ST-Correlation), we group road segments for decompose the whole road network. Accordingly, this mining module consists of 1) Marginal Velocity Estimation, 2) Spatial-temporal Centrality Estimation, and 3) Road Network Decomposition.

4.1 Marginal Velocity Estimation

To our best knowledge, existing map-matching methods always consider the speed limitation of road for measuring the probability of matching a trajectory to a road segment. However, people always do not drive so fast that the velocities of most vehicles could not achieve the speed limitation of road. Besides, Accordingly, the traditional map-matching methods probably match a low speed vehicle's GPS point to a low speed street although the vehicle runs along a highway. In other word, the traditional map-matching methods cannot work well based on the metrics from the speed limitation of road. Therefore, we propose a novel velocity estimation instead of the road segment's speed limitation. To reasonably estimate the road segment's speed, the dynamic of road segment is necessary to be considered. Fortunately, the real-time traffic time of each path is easily crawled from Google Maps. As a result, in this section, a real-time vehicle speed metric is proposed based on the traffic time provided by Google Maps.

Before introducing the real-time vehicle speed metric, we first describe the formal definitions for illustrating the real-time speed metric:

Definition 1. (Road Segment) Given a road network G, the road network G could be represented by a directed graph (i.e., $G = (V, E)$), and a vertex $v \in V$ represents an intersection point of a road or a corner of a road. A road segment is an edge $e \in E$. Each edge is assigned two real number vectors for representing the dynamics of road segment.

Definition 2. (Google Traffic Time) Given a certain timestamp ct and a road segment e, the temporal condition tc, the predecessor $p \in V$, and the successor $s \in V$ is utilize as the input (i.e., start point and destination) of routing service of Google Maps. Under temporal condition tc, the road segment e's Google Traffic Time $GT\text{-}Time(e, ct)$ is the traffic time which is provided by the routing service.

Definition 3. (Marginal Google Traffic Time) Given a road segment e and a time slot ts of a day, such as 17:00~19:00, the Marginal Google Traffic Time of Weekdays (denoted as $MGT_W(e, ts)$) is the minimum of Google Traffic Time of the time slot ts during weekdays, and the Marginal Google Traffic Time of Holidays (denoted as $MGT_H(e, ts)$) is the minimum of Google Traffic Time of the time slot ts during holidays or a weekends.

Table 1 shows an example of crawled google traffic time of a road segment e. We can observe that we separate one day into three time slots, 0:00 to 8:00, 8:00 to 16:00, and 16:00 to 24:00. According to the Table 1, the $MGT_W(e, 0{:}00 \sim 8{:}00) = 110$ (sec.), because the minimum of Google Traffic Time of the first time slot, 0:00 ~ 8:00, during weekdays is 110 seconds. According to Marginal Google Traffic Time under different time slots, we can easily estimate the possible maximal velocity of vehicles running along the given road segment. As mention in the section of Introduction, Google Maps crawled many vehicles' GPS logs and use the GPS logs to estimate the current traffic so that the traffic time could be viewed as a margin of the road segment's traffic time. In other word, most vehicles runs along the road segment would be slower than or approach to the speed, called *Marginal Velocity (M-Velocity)*. Given a road segment e and a time slot k, the M-Velocity is formally defined as follows:

$$M\text{-}Velocity(e, k) = \begin{cases} \dfrac{length\ of\ e}{MGT_W(e, k)}, & \text{if the day is weekday} \\ \dfrac{length\ of\ e}{MGT_H(e, k)}, & \text{otherwise} \end{cases} \quad (1)$$

Table 1. Crawled Google Traffic Time of a Road Segment

Date	*0:00 ~ 8:00*	*8:00 ~ 16:00*	*16:00 ~ 24:00*
2014/1/4 (Sat.)	**110** (sec.)	147 (sec.)	**160** (sec.)
2014/1/5 (Sun.)	184 (sec.)	**101** (sec.)	200 (sec.)
2014/1/6 (Mon.)	134 (sec.)	64 (sec.)	**91** (sec.)
2014/1/7 (Tue.)	120 (sec.)	**48** (sec.)	206 (sec.)
2014/1/8 (Wed.)	**77** (sec.)	61 (sec.)	197 (sec.)
2014/1/9 (Thur.)	115 (sec.)	58 (sec.)	209 (sec.)
2014/1/10 (Fri.)	111 (sec.)	90 (sec.)	105 (sec.)

We believe that the vehicles' speed can reflect some dynamics of a road segment. As mentioned earlier, the dynamics of road segments could be considered an important factor which influence each driver's decision of driving route. Therefore, we can use the vehicles' speed to improve accuracy map-matching methods. As a result, one of vectors of dynamic could be filled with the value of M-Velocity.

4.2 Spatial-temporal Centrality Mining

In this section, we propose and extract two kinds of centrality-based metric, including *spatial-temporal betweenness* (*ST-Betweenness*) and *spatial-temporal betweenness correlation* (*ST-Correlation*). Among them, the ST-Betweenness is used to understand the proportion of their common bridge road segments and the ST-Correlation is for measuring the correlation of each pair of road segments based on the ST-Betweenness.

4.2.1 Spatial-temporal Betweenness

Before introducing the ST-Betweenness, we first describe the conventional betweenness centrality. The betweenness centrality measures the degree of connectivity of a node (i.e., an intersection of roads in this paper) in a network. Given a graph $G = (V, E)$, the betweenness centrality of $v \in V$ is formally defined as follows:

$$Betweenness(v) = \sum_{s \neq v \neq t \in V} \frac{\#\text{of shortest path from } s \text{ to } t \text{ through } v}{\#\text{of shortest path from } s \text{ to } t} \quad (2)$$

As mentioned earlier, the spatial-temporal property should be considered for analyzing road network. Moreover, the shortest path may not be a fast or popular path. When we consider the spatial-temporal property for analyzing dynamic of road segment, the inherent modification is to replace shortest path by taxis' trajectory. As argued by Yuan *et al.* [21], taxi drivers could be viewed as experienced drivers. As the result, the taxi trajectory can bring the real dynamics of road segment into our analysis task. Therefore, we could modify the traditional betweenness centrality as spatial-temporal betweenness (ST-Betweenness).

Before introducing the real-time vehicle speed metric, we first describe the formal definitions for illustrating the real-time speed metric:

Definition 4. (Trajectory Set with Spatial Constraint) Given three road segments e_1, e_2 and e_3, the Trajectory with Spatial-

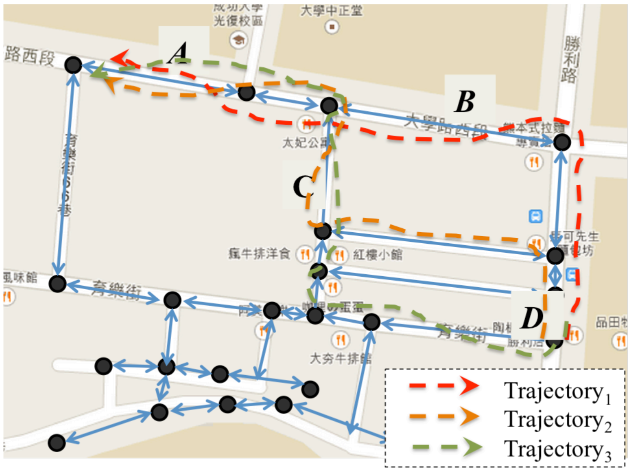


Figure 3. An Example of Trajectory Set

temporal Constraint (denoted as $STr(e_1, e_2, e_3)$) is the set of trajectory which is from e_1 to e_2 via e_3.

Take Figure 3 as an example. We can observe that the road segment B is passed by Trajectory$_1$ form road segment D to road segment A. Both Trajectory$_2$ and Trajectory$_3$ pass along the road segment C form road segment D to road segment A. Accordingly, $STr(D, A, B)$ is {Trajectoy$_1$}, and $STr(D, A, C)$ is {Trajectory$_2$, Trajectory$_3$}.

Definition 5. (Trajectory Set with Temporal Constraint) Given a time slot *ts* of a day, such as 17:00~19:00, the Trajectory Set with Temporal Constraint (denoted as $TTr(ts)$) is the set of trajectory of which timestamp of start point belongs to *ts*.

Take Table 2 and Figure 3 as an example. Table 2 shows the timestamp of start point of each trajectory in Figure 3. Accordingly, TTr(6:00 ~ 8:00) is {Trajectoy$_2$}, and TTr(17:00~19:00) is {Trajectory$_1$, Trajectory$_3$}.

Table 2. Timestamp of start point of each trajectory

Trajectory	***Timestamp of start point***
Trajectory$_1$	17:23:51
Trajectory$_2$	07:28:16
Trajectory$_3$	18:41:23

According to trajectory set with spatial constraint and temporal constraint, we can easily determine the spatial-temporal centrality of a given road segment. As mentioned earlier, most taxi driver is very experienced such that they would try to avoid a bridge road segment in their route. Here, the bridge road segment is the road segment which connects two disconnected sub road networks. Therefore, their routes can reflect the best cases of dynamic of road segment. In other word, the bridge road segment which is passed by the route must be necessary for connecting the source and destination of the route. Accordingly, we utilize taxi trajectory for estimating Spatial-temporal centrality, called *Spatial-temporal Betweenness (ST-Betweenness)*. Given a road segment *e* and a time slot *k*, the ST-Betweenness is formally defined as follows:

$$ST\text{-}Betweenness(e,\ k) = \frac{\left|\bigcup_{s \neq e \neq t \in E} STr(s,t,e) \cap TTr(k)\right|}{\left|TTr(k)\right|} \quad (3)$$

Based on our proposed spatial-temporal betweenness, every road segment could be evaluated the betweenness centrality in spatial-temporal way. Then we could utilize the temporal condition for analyzing the road segment dynamics.

4.2.2 Spatial-temporal Betweenness Correlation

Since a path of a route is composed by many road segments, the relation among the dynamics of road segments plays a crucial role for map-matching task. As the result, relation of dynamic of each pair of road segments could be utilized for improving the map-matching method. To do so, we can use the correlation of the ST-Betweenness of each pair of road segments for representing their relation of dynamic. As mentioned earlier, our proposed ST-Betweenness is sufficiently able to reflect the dynamic of road segment. Thus, considering the ST-Betweenness correlation is sufficiently able to reflect the relation of dynamic of road segment. The *ST-Betweenness correlation* (*ST-Correlation*) is formally defined as follows:

$$ST\text{-}Correlation(e,\ e') = \frac{n\left(X(e)\bullet X(e')\right) - \|X(e)\|_1 \|X(e')\|_1}{\sqrt{(n-1)\left(\|X(e)\|_2\right)^2 - \left(\|X(e)\|_1\right)^2}\sqrt{(n-1)\left(\|X(e')\|_2\right)^2 - \left(\|X(e')\|_1\right)^2}}, \quad (4)$$

where $X(e)$ = <*ST-Betweenness*(*e*, 0:00~1:00), …, *ST-Betweenness*(*e*, 23:00~24:00)> which indicates the vector of *ST-Betweenness* of *e*, *n* indicates number of dimensions of the vector of *ST-Betweenness*, and $\|x\|_p$ indicates the *p*-norm [26] of *e*.

4.3 Modular Road Network Detection

As mentioned earlier, modularity optimization [15] provides a good solution for divide a network in to several sub-networks according to the strength of division of the network into the sub-networks. In other word, the traditional modularity optimization trends to divide a network such that all sub-network are highly connected, and connectivity between each pair of sub-network is as low as possible. Therefore, estimation of connectivity plays crucial role in the modularity optimization. Accordingly, the traditional modularity optimization is always given an objective function Q shown as formula (5) which is utilized to address the quality of modularity structures.

$$Q = \frac{1}{2m}\sum_{ij}\left[(A_{ij} - P_{ij}) \times \delta(C_i, C_j)\right] \quad (5)$$

where the sum runs over all pairs of vertices, A is the adjacency matrix, m is the total number of edges of the graph, and the $\delta(C_i,C_j)$ is an indicator function which indicates whether i and j are in the same module. The P_{ij} represents the excepted number of edges between vertices i and j in the null model. Here, the null model is a probabilistic model for estimating the expectation of number of edge(s) which links i and j. In other word, the P_{ij} can reflect the how strong the two vertices trend to be connected. Furthermore, the matrix P also can influence the result of modularity optimization.

As mentioned in the section of Introduction, the spatial-temporal dynamic of each road segment play important role for decomposing road network. In addition, the ST-Betweenness can reflect the spatial-temporal centrality of each road segment, and the M-Velocity is able to reflect the margin of the road segment's traffic time. Therefore, we can utilize both the ST-Betweenness and the M-Velocity to estimate the null model of modularity optimization task. Before introducing the definition of the spatial-temporal dynamic based on our proposed ST-Betweenness and M-Velocity, we first describe the formal definitions for illustrating

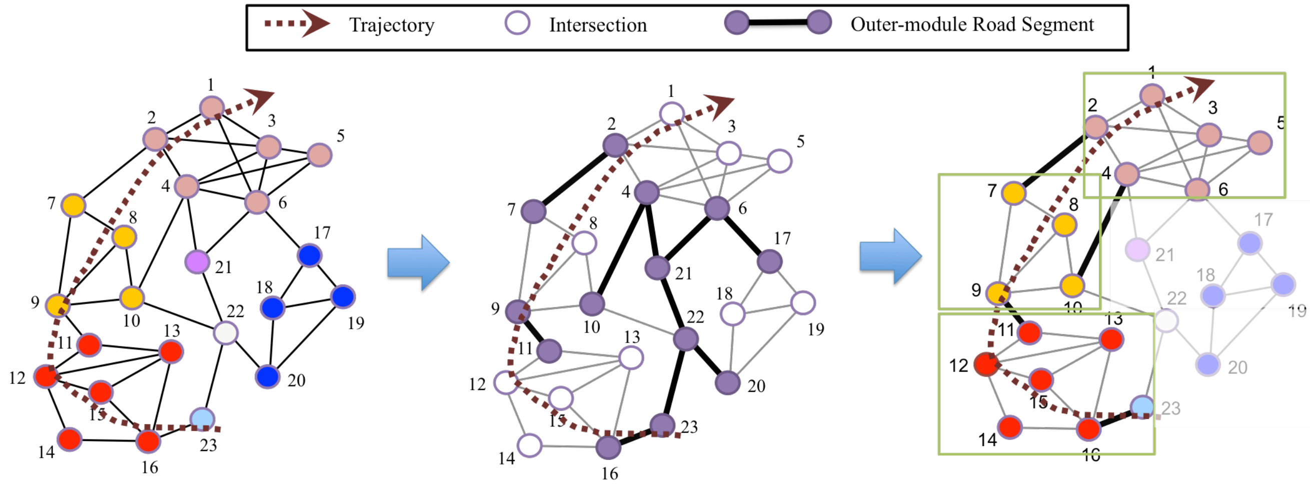


Figure 4. An Example of Two-phase Map Matching

the spatial-temporal dynamic based on our proposed ST-Betweenness and M-Velocity:

Definition 6. (Average M-Velocity) Suppose we divide a day into n time slots $\{ts_1, ts_2, ts_3, ..., ts_n\}$ (i.e., the dimension of the dynamic vector is n). Given a road segment e, the Average M-Velocity is formally defined as follows:

$$Average\ M\text{-}Velocity(e) = \frac{1}{n}\sum_{k=1}^{n} M\text{-}Velocity(e,\ ts_k) \qquad (6)$$

Definition 7. (Average ST-Betweenness) Suppose we divide a day into n time slots $\{ts_1, ts_2, ts_3, ..., ts_n\}$ (i.e., the dimension of the dynamic vector is n). Given a road segment e, the Average ST-Betweenness is formally defined as follows:

$$\frac{1}{n}\sum_{k=1}^{n} ST\text{-}Betweenness(e,\ ts_k) \qquad (7)$$

Accordingly, we formally define the spatial-temporal dynamic based on our proposed ST-Betweenness and M-Velocity. Given a road network $G = (V, E)$, the spatial-temporal dynamic matrix ($D_{|V|\times|V|}$) is shown as follows:

$$D[i,j] = (Average\ M\text{-}Velocity(e_{i,j}))^{Average\ ST\text{-}Betweenness(e_{i,j})}, \qquad (8)$$

where the $e_{i,j}$ is the edge which indicates the vertex v_i link to v_j. As mentioned earlier, the spatial-temporal dynamic of each road segment can be utilized for decomposing road network. As the result, we can treat the spatial-temporal dynamic matrix as the matrix P shown in the formula (5). To decompose the whole road network, we perform the modularity optimization algorithm [15]. As the result, the whole road network can be decomposed into several small road networks. The road segments that connect different sub-networks are called outer-module road segments, and others are called inner-module road segments.

5. THE TWO-PHASE MAP MATCHING

With the spatial-temporal metrics and modularity of road network, we are able to describe our two-phase map-matching method. The first phase, called module-matching phase, is to divide the GPS trajectory into several sub-trajectories according to the result of decomposition of road network. The second phase, called inner-module map-matching phase, is to match each sub-trajectory onto the road in each small road network as shown in Figure 4.

5.1 Module-Matching Phase

In the module-matching phase, we focus only on the outer-module road segments. We try to match the input trajectory onto the outer-module road segments. Since the network size of outer-module road segments should be relatively small compared with that of whole network, the density of the network of outer-module road segments should be much lower. Therefore, we can use straightforward method for dealing with this task, such as the geometric-distance-based matching. For each GPS point of the input trajectory, we match it onto the nearest intersection of road, i.e., we directly match a GPS point on the nearest vertex which is predecessor or successor of an outer-module road segment. As shown in Figure 4(b), the GPS points of the trajectory should be matched on the vertices 2, 7, 9, 11, 16, and 23.

5.2 Inner-Module Map-Matching Phase

After module-matching phase, we perform a global map-matching algorithm, proposed by Lou *et al.* [14], on a sub-network and use the part of the GPS trajectory which belong to the sub-network as the input of the algorithm as shown in Figure 4(c). In the map-matching algorithm, the core idea is to define a matching score which can represent the goodness of matching the GPS trajectory onto a candidate path. In [14], the top k nearest road segments of a GPS point is selected to form candidate set of GPS point. Accordingly, a candidate graph G' (V', E') is able to be generated for input trajectory. Given a trajectory T: $p_1 \rightarrow \ldots \rightarrow p_n$. V' is a set of candidate points for each GPS point, and E' is a set of edges representing the shortest paths between any two neighboring candidate points, as shown in Figure 5. As the result, map-matching problem can be formulated as the maximum score path searching proem, and the definition of matching score can be estimated by the sum of the score of edge of path.

As mentioned earlier, the road dynamic can fully be considered only if using our proposed spatial-temporal metrics, i.e., ST-Betweenness, ST-Correlation and M-Velocity. Thus, with the spatial and temporal mining above, we are ready to modify the matching score of Lou *et al.*'s map-matching algorithm used in the result matching component. For each edge $x \rightarrow y$ of the candidate graph, our scoring function is defined as follows:

$$F(x \rightarrow y) = GeoScore(y) \times CorrScore(x, y), \qquad (9)$$

where the *GeoScore*(y) represents the how geographically close between the GPS point and its candidate road segment y, and the *CorrScore*(x, y) indicates how similar between the two candidate road segments, x and y. Accordingly, we formally define the *GeoScore*(y) and the *CorrScore*(x, y) based on ST-Betweenness, ST-Correlation and M-Velocity, respectively.

Before introducing the *GeoScore*, we first describe the formal definitions for illustrating the *GeoScore*:

Definition 8. (Reference GPS Point of a Candidate Road Segment) Given a Candidate Road Segment y, the Reference GPS point of y denoted as $Gp(y)$ is the GPS point of which candidate set contain the road segment y.

Based on the definition of reference GPS point of a candidate road segment, the *GeoScore* is formally defined as follows:

$$GeoScore(y) = \int_{r}^{\infty} \frac{1}{S(y)\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x}{S(y)}\right)^2} dx, \qquad (10)$$

where r is the Euclidean distance between the GPS point and its candidate road segment y, and the $S(y)$ indicates the dynamic error of GPS. We believe that the GPS error is proportional to the road segment's betweenness and marginal speed. Formally, we define the dynamic error $s(y)$ as follows:

$$S(y) = M\text{-}Velocity(y,k)^{ST\text{-}Betweenness(y,k)}, \qquad (11)$$

where k is the time slot determined by the $Gp(y)$.

The core idea of *GeoScore* is to estimate the how geographically close between the GPS point and its candidate road segment y by normal distribution with 0 mean and $S(y)$ standard deviation. We use the probability $P[X > r]$ of a normal distribution to determine the *GeoScore*.

Before introducing the *CorrScore*, we first describe the formal definitions for illustrating the *CorrScore*:

Definition 9. (M-Velocity of a Path) Given a path from the road segment x to the road segment y, the M-Velocity of the Path is formally defined as follows:

$$\mu(x, y) = \frac{1}{n}\sum_{k=1}^{n} M\text{-}Velocity(e_k,\ ts), \qquad (12)$$

Definition 10. (Variance of M-Velocity of a Path) Given a path from the road segment x to the road segment y, the variance of M-Velocity of the Path is formally defined as follows:

$$\sigma^2(x, y) = \frac{1}{n}\sum_{k=1}^{n}\left(M\text{-}Velocity(e_k,\ ts) - Path\ M\text{-}Velocity(x, y)\right)^2, \qquad (13)$$

Definition 11. (Sampling Average Velocity of a Path) Given a path from the road segment x to the road segment y, the Sampling Average Velocity of the Path is formally defined as follows:

$$sv(x, y) = \frac{dist(Gp(x), Gp(y))}{\Delta t(Gp(x), Gp(y))} \qquad (14)$$

where the $dist(*,*)$ indicates the Euclidean distance and $\Delta t(*,*)$ indicates the time interval between the two GPS points.

Definition 12. (Standard difference between Sampling Velocity and M-Velocity) Given a path from the road segment x to the road segment y, the Standard difference between Sampling Velocity and M-Velocity is formally defined as follows:

$$d(x, y) = \frac{sv(x, y) - \mu(x, y)}{\sqrt{\sigma^2(x, y) / n}}, \qquad (15)$$

where the path $x \rightarrow y = \{x = e_1, e_2, e_3, \ldots, e_n = y\}$.

Based on the above definitions, the *CorrScore* is formally defined as follows:

$$CorrScore(x, y) = \left(\int_{d(x,y)}^{\infty} \frac{1}{\sqrt{n}B(\frac{1}{2}, \frac{n}{2})}\left(1 + \frac{v^2}{n}\right)^{-\frac{n+1}{2}} dv\right)^{-ST\text{-}Correlation(x,y)}, \qquad (16)$$

where n is the length of the shortest path from x to y, and $B(*,*)$ is the beta function [24]. The core idea of *CorrScore* is to estimate the how similar between the two candidate road segments, x and y, in spatial-temporal way. We believe that the similarity between two candidate road segments is proportional to the deference between the Sampling Average Velocity and M-Velocity of the shortest path communicating the two candidate road segments. Thus, we adopt the one-sample t-test to evaluate the theoretical deference between the Sampling Average Velocity and M-Velocity of the Path. The physical mean of the radix part is the p-value of the one-sample t-test [27]. Besides, we utilize the ST-Correlation to control the power of the p-value. Since the p-value is probability, it belongs to the interval from 0 to 1. Thus, the power part of the *CorrScore* is set as –ST-Correlation.

According to *GeoScore* and *CorrScore*, our scoring function of each edge of the candidate graph is well defined. As mentioned earlier, the score of a path in the candidate graph is the summation of all the scores of edges of the path. Therefore, the map-matching result can be computed by searching a path with maximum score in the candidate graph. To search the path with maximum score, we perform the FindMatchedSequence algorithm that is proposed by Lou *et al.* [14].

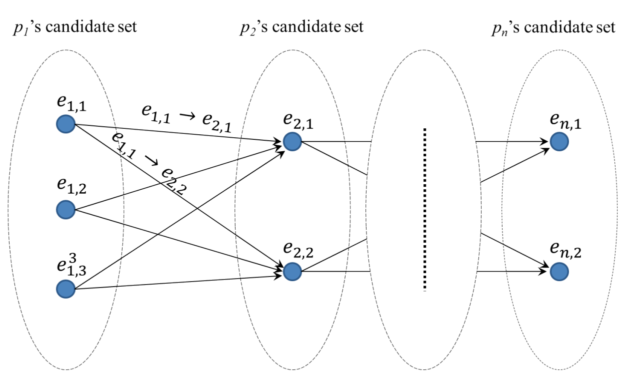


Figure 5. An Example of Candidate Graph

6. EXPERIMENTS

In this section, we conduct a series of experiments to evaluate the performance for the proposed *UrbMatch* using the real road network of Shanghai collected from Open Street map [25] and a real trajectory set obtained from WnSN, SJTU [28]. All the experiments are implemented in Java JDK 1.6 on an Intel Xeon CPU W3520 2.67GHz machine with 24GB of memory running Microsoft Windows win7. We first describe the data preparation on the real road network of Shanghai collected from Open Street map [25] and a real trajectory set obtained from

WnSN, SJTU [28]. Then, we introduce the evaluation methodology. Finally, we show our experimental results for following discussions.

6.1 Evaluation Dataset and Methodology

The road network of Shanghai collected from Open Street map [25]. The road network consists of 65,882 road segments and 48,838 intersections. That is, if we use a directed graph $G(V,E)$ to represent it, the size of vertex set V is 48,838, and the size of edge set E is 65,882. We use the trajectory data from GPS-equipped taxis of Shanghai city, which were collected by Wireless and Sensor networks Lab (WnSN), Shanghai Jiao Tong University [28]. However, the trajectories are not labeled true path, i.e., there is no ground truth for map-matching task. Thus, the trajectory data from GPS-equipped taxis of Shanghai city is only used for ST-Miner. As the result, we use a vehicle movement simulation model [12] to form the testing dataset of trajectory.

The follows are the main measurements for the experimental evaluations. The *Accuracy by Number* (*AN*) and *Accuracy by Length* (*AL*) are defined as Equations (17) and (18).

$$AN = \frac{\#\,correctly\ matched\ road\ segments}{\#\,all\ road\ segments\ of\ the\ trajectory} \tag{17}$$

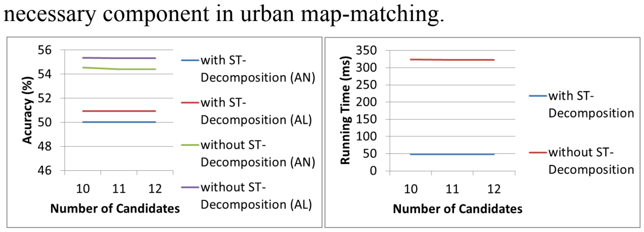
$$AL = \frac{total\ length\ of\ correctly\ matched\ road\ segments}{total\ length\ of\ all\ road\ segments\ of\ the\ trajectory} \tag{18}$$

6.2 Experimental Results

We divide the experiment into two parts: 1) Impact of Road Network Decomposition and 3) Comparison of Existing Map-Matching Method. We examine the impact of Road Network Decomposition in terms of running time, Accuracy by Number and Accuracy by Length. For the comparison of existing map-matching method, we compare our method with ST-Matching [14] in terms of Accuracy by Number and Accuracy by Length.

6.2.1 Impact of Road Network Decomposition

Figure 6 (a) shows the Accuracy by Number and Accuracy by Length of our map-matching method with/without road network decomposition under various sizes of candidate set. We can observe that all results of our map-matching method without road network decomposition are slightly accurate than that with road network decomposition. The reason is that the method without road network decomposition consider whole road network for map-matching task. Therefore, it would be more accurate. However, it just achieves about 5% accuracy higher than the method with road network decomposition. But, if we consider the running time shown in Figure 6(b), we can find that such tiny accuracy gain would cost about 6 times as much running time. Accordingly, our road network decomposition is still a useful and necessary component in urban map-matching.



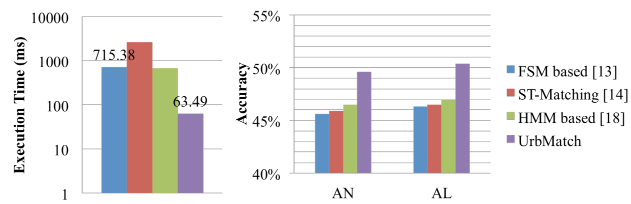
(a) (b)

Figure 6. Impact of Road Network Decomposition under Various Sizes of Candidate Set

Besides, we also can see that the size of candidate set would not affect our method. The reason is that our spatial-temporal information is mined from historical trajectory dataset. It can easily distinguish the true path form the candidate set.

6.2.2 Comparison of Existing Map-Matching Method

Figure 7 (b) shows the Accuracy by Number and Accuracy by Length of our map-matching method and several state-of-the-art map-matching methods, Liu *et al.*'s Finite-State-Machine-based Map-matching method [13], Lou *et al.*'s ST-Matching [14] and Raymond *et al.*'s Hidden-Markov-Model-based Map-matching method [18], under various sizes of candidate set. We can observe that all results of our map-matching method are slightly accurate than ST-Matching. The reason is that the all these state-of-the-art map-matching methods partially consider spatial-temporal information to help improve accuracy of map-matching task. Our map-matching method fully and deeply addresses spatial-temporal information mining for improving accuracy of map-matching task. Therefore, our method would be more accurate than ST-Matching which is one of state-the-art global map-matching mehtod.



(a) (b)

Figure 7. Comparison of Existing Map-Matching Methods

Besides, as shown in Figure 7(a), we can find that our method is eleven times faster than Finite-State-Machine-based Map-matching method [13] which is state-of-the-art local map-matching method. Although the local map-matching methods are efficiency-oriented, our UrbMatch significantly outperforms the Finite-State-Machine-based Map-matching method [13]. The reason is that our method involves the road network decomposition component.

7. CONCLUSIONS & FUTURE WORK

In this paper, we propose a new multi-track map-matching algorithm called Urban Map-Matching (*UrbMatch*) to match urban GPS data onto a digital map of a city. The map-matching method fully and deeply addresses spatial-temporal information mining for improving accuracy of map-matching task. Based on mined spatial-temporal information, we decompose the whole road network for speeding up the map-matching task. We propose a two-phase map-matching algorithm which employs decomposed road network and mined spatial-temporal information to generate a candidate graph, from which a sequence of matched results with highest sum of score is identified as the matching result. The experiment results demonstrate that our *UrbMatch* significantly outperforms state-of-the-art map-matching method, ST-Matching [21], in terms of accuracy and running time. In our future work, we plan to deal with the problem that multi-track map-matching algorithm is sensitive to the GPS sampling rate.

Acknowledgement

This research was partially supported by Ministry of Science and Technology, Taiwan, R.O.C. under grant no. 101-2221-E-006-255-MY3.

REFERENCES

[1] H. Alt, A. Efrat, G. Rote and C. Wenk. Matching planar maps. J. of Algorithms, 49:262–283, 2003.

[2] H. Alt and M. Godau. Computing the Fréchet distance between two polygonal curves. Int. J. Comput. Geom. Appl., 5:75–91, 1995.

[3] S. Brakatsoulas, D. Pfoser, R. Salas and C. Wenk, 2005. On map-matching vehicle tracking data. In Proceedings of the 31st international Conference on Very Large Data Bases, 853-864, 2005.

[4] S.S. Chawathe. Segment-based Map Matching. In IEEE Symposium on Intelligent Vehicles, 2007.

[5] T. Cheng, J. Haworth and J. Wang. Spatio-temporal autocorrelation of road network data. Journal of Geographical Systems, 2011.

[6] A. Comber, C. Brunsdon and E. Green. Using a GIS-based network analysis to determine urban greenspace accessibility for different ethnic and religious groups. Landsc Urban Plan, 86(1):103-114, 2008.

[7] J. Greenfeld. Matching GPS observations to locations on a digital map. In Proceedings of 81th Annual Meeting of the Transportation Research Board, 2002.

[8] B. Hillier and S. Iida. Network effects and psychological effects: A theory of urban movement", in van Nes (2005), pp 553-564, 2005.

[9] B. Hummel and K. Tischler. GPS-only map matching: exploiting vehicle position history, driving restriction information and road network topology in a statistical framework. In Proceedings of the GIS Research UK Conference, 68–77, 2005.

[10] A. Javanmard, M. Haridasan and L. Zhang. Multi-track map matching. In *Proc. 20th International Conference on Advances in Geographic Information Systems (GIS '12)*, pages 394--397, 2012.

[11] J. Krumm and E. Horvitz. The Microsoft multiperson location survey. Technical Report MSR-TR-2005-103, Microsoft Research, Redmond, WA, USA, 2005.

[12] K.-C. Lan and Chien-Ming Chou, "Realistic Mobility Models for Vehicular Ad hoc Network (VANET) Simulations," in IEEE ICST 2008, October 2008

[13] S. Liu, Z. Shi, M. Zhao, W. Xu, K. Zhang. An Urban Map Matching Algorithm Using Rough Sensor Data. In Proceedings of PEITS'08, 2008.

[14] Y. Lou , C. Zhang , Y. Zheng , X. Xie , W. Wang , Y. Huang. Map-matching for low-sampling-rate GPS trajectories, Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, November 04-06, 2009, Seattle, Washington

[15] E. L. Martelot and C. Hankin. Fast Multi-Scale Detection of Relevant Communities in Large Scale Networks. The Computer Journal, Oxford University Press, 2013.

[16] P. Newson and J. Krumm. Hidden Markov map matching through noise and sparseness. In GIS, pages 336–343, 2009.

[17] O. Pink and B. Hummel. A statistical approach to map matching using road network geometry, topology and vehicular motion constraints. In the 11th International IEEE Conference on Intelligent Transportation Systems, 2008.

[18] R. Raymond, T. Morimura, T. Osogami and N. Hirosue. Map Matching with Hidden Markov Model on Sampled Road Network. In Proceedings of ICPR'12, 2012.

[19] C. Wenk, R. Salas and D. Pfoser. Addressing the need for map-matching speed: localizing global curve- matching algorithms. In Proceedings of the 18th international Conference on Scientific and Statistical Database Management, 2006.

[20] H. Yin and O. Wolfson. A Weight-based map matching method in moving objects databases. In Proceedings of the 16th international Conference on Scientific and Statistical Database Management, 437, 2004.

[21] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie and Y. Huang. T-Drive: Driving Directions Based on Taxi Trajectories. In *Proceedings of ACM SIGSPATIAL GIS*'10, 2010.

[22] J. Yuan , Y. Zheng , C. Zhang , X. Xie , G.-Z. Sun, An Interactive-Voting Based Map Matching Algorithm, *Proceedings of the 2010 Eleventh International Conference on Mobile Data Management*, p.43-52, May 23-26, 2010

[23] Y. Zheng, L. Capra, O. Wolfson, H. Yang. Urban Computing: concepts, methodologies, and applications. *ACM Transaction on Intelligent Systems and Technology (ACM TIST)*. 5(3), 2014

[24] Beta function: http://en.wikipedia.org/wiki/Beta_function

[25] Open Street Map: http://www.openstreetmap.org/.

[26] P-norm: http://en.wikipedia.org/wiki/P-norm

[27] Student's t-test: http://en.wikipedia.org/wiki/Student's_t-test

[28] SUVnet-Trace data: http://wirelesslab.sjtu.edu.cn.