Transportation Research Record

Online Map Matching Based on Higher-Order Hidden Markov Model

--Manuscript Draft--

Full Title:	Online Map Matching Based on Higher-Order Hidden Markov Model
Abstract:	Map matching is a key pre-process of trajectory data which recently have become a major data source for various transport applications and location-based services. In this paper, an online map matching algorithm based on higher-order hidden Markov model (HMM) is proposed for processing trajectory data in complex urban road network environment such as parallel road segments and various road intersections. Various factors such as driver's travel preference, network topology, road level and vehicle heading are considered. An extended Viterbi algorithm is developed to solve the map matching problem efficiently, and a self-adaptive sliding window mechanism is proposed to adjust window size on a real time basis. To demonstrate the effectiveness of the proposed algorithm, a case study is carried out using a massive taxi trajectory dataset in Nanjing, China. Case study results show that the proposed algorithm outperforms state-of-the-art algorithms built on the first-order HMM in various testing environment.
Manuscript Classifications:	Data and Information Technology; Geographic Information Science and Applications ABJ60; Geographic Information Systems; GPS Data
Manuscript Number:	20-01847
Article Type:	Presentation
Order of Authors:	Xiao Fu
	Jiaxu Zhang
	Yue Zhang
	Biyu Chen
	Zhiyuan Liu

Online Map Matching Based on Higher-Order Hidden Markov Model

Xiao Fu
Associate Professor
ORCID: 0000-0003-0446-0971
Jiangsu Key Laboratory of Urban ITS, Jiangsu Province Collaborative Innovation Center of Modern Urban Traffic Technologies, School of Transportation
Southeast University, Nanjing, China, 211189
Email: fuxiao@seu.edu.cn

Jiaxu Zhang
Research Scholar
School of Transportation, Southeast University
Nanjing, China, 211189
Email: z.jiaxu@foxmail.com

Yue Zhang
Research Scholar
School of Transportation, Southeast University
Nanjing, China, 211189
Email: zhangy-1021@foxmail.com

Biyu Chen
Professor
State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University
Wuhan, China, 430000
Email: chen.biyu@whu.edu.cn

Zhiyuan Liu (Corresponding Author)
Professor
ORCID: 0000-0002-6331-0810
Jiangsu Key Laboratory of Urban ITS, Jiangsu Province Collaborative Innovation Center of Modern Urban Traffic Technologies, School of Transportation
Southeast University, Nanjing, China, 211189
Email: zhiyuanl@seu.edu.cn

Word Count: 6392 words + 2 tables = 6892 words

Submitted [1 August 2019]

ABSTRACT

Map matching is a key pre-process of trajectory data which recently have become a major data source for various transport applications and location-based services. In this paper, an online map matching algorithm based on higher-order hidden Markov model (HMM) is proposed for processing trajectory data in complex urban road network environment such as parallel road segments and various road intersections. Various factors such as driver's travel preference, network topology, road level and vehicle heading are considered. An extended Viterbi algorithm is developed to solve the map matching problem efficiently, and a self-adaptive sliding window mechanism is proposed to adjust window size on a real time basis. To demonstrate the effectiveness of the proposed algorithm, a case study is carried out using a massive taxi trajectory dataset in Nanjing, China. Case study results show that the proposed algorithm outperforms state-of-the-art algorithms built on the first-order HMM in various testing environment.

Keywords: Map Matching, Higher-Order Hidden Markov Model, Trajectory Analysis

INTRODUCTION

With the development of positioning and wireless communication technologies, floating car data (e.g. trajectories of taxis) have become major data source for many applications such as location-based services (LBS) (*1*), intelligent transportation systems (ITS) (*2*) and transport policy appraisals (*3*). The errors of data collected by global positioning system (GPS) equipment on floating vehicles are inevitable which come from satellite, transmission process and receiver (*4*). Map matching is the process of matching GPS data with errors onto road network, which can eliminate the impact of errors and maximize the effectiveness of data. The matched GPS data provide abundant traffic information and drivers' behaviour information for scientific research and applications.

Existing map matching algorithms can be divided into four categories based on the technique they adopted (*5*): geometric technique (*6*, *7*), topological technique (*8*, *9*), probability statistics technique (*10*) and other advanced techniques (*11*, *12*). The algorithms with geometric technique utilize geometric information of GPS point and road network (e.g. distance, angle and shape) without considering the topology of road network. These algorithms show high efficiency of map matching, but the accuracy is low when matching low-precision GPS data to complex road network. As regards topological technique, both geometric factors and road topology are considered. To some extent, topological technique improves the matching accuracy, but is still vulnerable to the influence of low-frequency sampling interval and large sampling noise. The probability statistics technique sets an ellipse or rectangle confidence area for each GPS point, thus we can obtain the probability according to the distance between the GPS point and the position in confidence area. Optimal matching paths are determined according to values of the probability. Compared to geometric technique and topological technique, the probability statistics technique is relatively complex and difficult to implement, and shows low time efficiency. Combining geometric, topological and probability factors, some advanced techniques such as Kalman filter (*13*), Bayesian filter (*14*), fuzzy logic model (*15*) and hidden Markov model (HMM) (*16*) can effectively improve the map matching accuracy and achieve online incremental matching.

Of the advanced techniques, recently, HMM has become popular in map matching studies. HMM is a prevailing paradigm of network-based dynamics modelling. Hidden Markov chain can generate random hidden state series, and each hidden state generates an observed state. This mechanism is consistent with the process of finding the most suitable matching point (hidden state) to each GPS point (observed state) on the road network in map matching problem. Existing map matching algorithms based on HMM can be categorized into two kinds (*14*): offline algorithm and online algorithm (refer to Table 1).

Offline HMM map matching algorithms use historical data, batching the whole input trajectory (*17-22*) to find the optimal matching path in the road network. Whole trajectory enables offline algorithms to take full account of the relationship between the front and the back points to achieve higher accuracy. Offline algorithms show robustness to the reduction of sampling rate, but the computation efficiency is low. Online algorithms estimate the current segment immediately after obtaining GPS data, and this kind of algorithm can be used for providing online services such as real-time navigation and trajectory monitoring. Because of the unavailability of future points, online algorithms are more complicated, but have higher computation efficiency and can satisfy the real-time need. Some studies (*23*, *24*) utilize sliding window with fixed window size to realize online matching. As GPS points increase, the points in sliding window change dynamically. However, under the condition of low data quality or complex road network, small window leads to a significant decrease in matching accuracy while large window brings a significant decrease in computation efficiency. Considering these, in this study, we proposed self-adaptive sliding window to realize online map matching based on HMM, which promises accuracy and efficiency at the same time.

HMM includes the stochastic process of observation and the stochastic process of state transition. In map matching algorithms based on HMM, two probabilities are of importance, i.e. observation probability and transition probability. Observation probability is normally obtained by the Gaussian distribution of large circle distance between GPS point and candidate point. Some factors can be considered in calculating observation probability. For instance, Unsupervised HMM (*20*) considers the location of Antenna when matching mobile phone data. Quick Matching (*21*), SnapNet (*23*) and Route Choice HMM (*24*) add more factors such as the speed constraints of different road levels, major road priority and vehicle heading. As regards transition probability calculation, to consider temporal relationship of different points, speed constraint is considered in several studies (*17*, *18*, *20*). To consider spatial relationship, some factors are included such as short route priority (*19*, *23*, *24*), segment direction change (*22*) and same road priority (*23*).

Based on the analysis of advantages of each algorithm, this study is a pioneering endeavour devoted to comprehensively considering various factors in online map matching, i.e. major road priority, vehicle heading, driver's travel preference, short route priority and same/adjacent road priority.

TABLE 1 Comparison of HMM-based Map Matching Studies

Category	Model	Sliding window	Factors of observation probability	Factors of transition probability	Order of HMM
Offline	Spatial and Temporal Matching (*17*)	×	×	Speed constraint	First-order
	Interactive-Voting Matching (*18*)	×	×	Speed constraint	
	HMM Matching (*19*)	×	×	Short route priority	
	Unsupervised HMM (*20*)	×	Antenna location	Speed constraint	
	Quick Matching (*21*)	×	Speed constraint	×	
	Multistage Matching (*22*)	×	Vehicle heading	Segment direction change	
Online	SnapNet (*23*)	Fixed size	Vehicle heading; Major road priority	Short route priority; Same road priority	
	Route Choice HMM (*24*)	Fixed size	×	Short route priority; Speed constraint	
	This Study	Self-adaptive size	Major road priority; Vehicle heading; Driver's travel preference	Short route priority; Same/adjacent road priority	Second-order

To the best of our knowledge, almost all map matching algorithms based on HMM adopt first-order HMM. The basic hypothesis of first-order HMM is that the observation probability is only related to the current state while the transition probability is only related to the previous state. Because the moving of a vehicle is a continuous process, there is a complex space-time relationship between the current state and the previous states. There is no doubt that first-order HMM over-simplifies several practical systems. Recently, Salnikov *et al*. explored possibilities to enrich the system description and exploited empirical pathway information by means of second-order Markov models (*25*). Experiments show that the higher order model is more effective than the first-order model in dealing with space-time continuum. Therefore, a need is likely to exist for solving map matching problem using higher-order (e.g. second-order) HMM to achieve better map matching results.

The merits of the proposed online map matching algorithm are summarized as follows. First, the proposed novel map matching algorithm is on the basis of higher-order HMM, which can better consider the space-time relationship among different states. It can be effectively applied to complex urban road network with parallel segments using low-frequency sampling GPS data. Second, driver's travel preference towards road segments is obtained by a self-learning algorithm. Road level, vehicle heading and other factors are taken into account when calculating the probability matrix of second-order HMM in order to improve the matching accuracy. Third, compared to conventional offline map matching algorithms, sliding window with self-adaptive size is proposed to achieve online incremental map matching.

The rest of this paper is organized as follows. In next section, we state the problem of map matching. After the problem statement, an online map matching algorithm is proposed based on second-order HMM. A case study is carried out using a large taxi trajectory dataset in Nanjing, China to test the validity of the algorithm under various road conditions. Finally, we conclude this study and discuss directions for further research.

PROBLEM STATEMENT

Vehicle trajectory data are a series of GPS points recorded in chronological order. Each GPS point indicates longitude and latitude, vehicle speed, timestamp, etc. Because the errors of data collected by GPS equipment are inevitable, map matching is a key process before using the vehicle trajectory data. It is a process of matching GPS data onto the road segments and obtaining the continuous and specific locations of vehicles on the road. The concepts used in this study are listed as follows:

GPS point*:* A GPS point g_t is a record indicating the longitude, latitude, time stamp and velocity of the vehicle.

GPS trajectory*:* A GPS trajectory T is a series of GPS points. A T is showed as: $g_1 \rightarrow g_2 \rightarrow \ldots \rightarrow g_n$.

Road network*:* Road network $G(V, E)$ is a directed graph where V is the set of vertexes and E is the set of edges.

Road segment*:* A road segment e is a directed edge in road network with length, road level, start vertex and end vertex.

Candidate point: The candidate point c_t^n is the nth candidate point matched with GPS point g_t on the road network.

Route: A route R is a sequence of road segments that matched best to a GPS trajectory T, each road segment belongs to the edge set E of road network $G(V, E)$. R is showed as: $e_1 \rightarrow e_2 \rightarrow \ldots \rightarrow e_n$.

With the above concepts, the map matching problem solved in this study can be defined as: Find the candidate points $c_t^1, c_t^2, \ldots, c_t^n$ on each road segment e corresponding to GPS point g_t. Select the most likely candidate points sequence for GPS trajectories T, and connect the matched road segments on network G to get route R.

HIGHER-ORDER HMM MAP MATCHING

Data Pre-processing

Generally, there are a lot of "redundancy" and "incompleteness" in floating vehicle GPS data, which may be caused by devices or road environments (e.g. stopping in or passing through tunnels). In order to ensure the efficiency and accuracy of map matching, we first need to pre-process the GPS data, including the removal of redundant data and the interpolation of missing data.

For the currently received data point g_t, calculate the large circle distance (*19*) of g_t and g_{t-1} (denoted as $D_{t-1,t}$), if $D_{t-1,t}$ is less than a pre-defined lower bound, the current point g_t is omitted and not matched. If $D_{t-1,\ t}$ is greater than an upper bound, the two points will be interpolated linearly.

With the data pre-processing, the redundant GPS data points can be effectively eliminated so as not to make unnecessary matching. At the same time, interpolation of two points with too large intervals helps to process low-frequency GPS data.

Candidate Point Selection

For the currently received data point g_t, we search for its candidate points (refer to Figure 1(a)) with the following steps:

Step1: Using the R-tree index, road segments within the range of a search radius and near to the point g_t are selected as candidate road segments.

Step2: Project g_t vertically on the candidate road segments, and the projection point c_t^i is a candidate point for g_t. If the projection point falls outside the segment, choose the closer vertex of the segment as c_t^i. As shown in Fig.1a, the candidate points for g_t are $c_t^1, c_t^2, \ldots, c_t^5$. The distances from g_t to the candidate points are denoted as $d_t^1, d_t^2, \ldots, d_t^5$ respectively.

Observation Probability

In first-order HMM, the observation probability is used to measure the probability of getting some kinds of observed value in a hidden state (*26*). The map matching algorithms based on HMM usually regard the GPS point g_t as the observation value of state t, and the actual position of g_t as the hidden value of state t. The observation probability is modelled using a Gaussian distribution for GPS trajectories (*23*). The first-order HMM observation probability in this paper is obtained as

$$P\left(g_t \mid c_t^i\right) = \frac{1}{\sqrt{2\pi}\sigma_t} e^{-0.5\left(\frac{\theta \cdot \rho \cdot d_t^i}{\sigma_t}\right)}, \tag{1}$$

where $P\left(g_t \mid c_t^i\right)$ is the observation probability of the candidate point c_t^i on g_t. d_t^i is the large circle distance between g_t and the candidate point c_t^i. σ_t is the standard deviation of a Gaussian random variable that corresponds to the average large circle distance between g_t and its candidate points. θ is a weight about vehicle heading, which is related to the road direction angle α_{raod} and the trajectory direction angle α_{GPS}:

$$\theta = \upsilon + \frac{e^{\left|\alpha_{road} - \alpha_{GPS}\right|}}{e^{2/\pi}}. \tag{2}$$

In Equation (2), the road direction angle α_{road} is the direction angle of the two vertexes of a segment. The trajectory direction angle α_{GPS} indicates the direction angle of the last GPS point and the current GPS point. Because of the bidirectional property of the road, there are two results of $\left|\alpha_{road} - \alpha_{GPS}\right|$, and the smaller value of the two results should be used. υ is a parameter which can be estimated with real data.

ρ is a weight about road considering the effect of road including road level (denoted as $rlevel$) and driver's travel preference for the road segment (denoted as $plevel$):

$$\rho = 1 - \mu\left(rlevel + plevel\right), \tag{3}$$

where μ is a parameter to be estimated. In this study, *rlevel* is within [0, 5]. A high *rlevel* indicates a high level of road. The value of *plevel* is also ranging from 0 to 5. Considering driver's travel experience as a sigmoid curve (*27*), *plevel* can be derived as

$$plevel = \frac{5}{1 + e^{-\varpi + \varpi'}}, \tag{4}$$

where ϖ is the number of times a driver drives through a road segment, and ϖ' is the number of trips the driver needs to make himself familiar with the road.

In this way, the observation probability can be obtained. By using vehicle heading weight θ and road weight ρ, we can consider road level, driver's travel preference and heading of the floating vehicle at that time, which are significant in online map matching with limited information. Take Figure 1(b) as an example to illustrate the merit of road weight ρ. The current GPS point g_t is located in the middle of two parallel road segments. The distances from g_t to c_t^1 and c_t^2 are the same. In conventional map matching methods, c_t^1 or c_t^2 is selected randomly as the real position of vehicle. However, if road level and travel experience are taken into account using our proposed method, we can consider c_t^1 as the real position of vehicle. It can be seen that without subsequent GPS points, we must make full use of the information provided by existing GPS points and road network in order to improve the matching accuracy.

Figure 1(c) shows the merit of incorporating vehicle heading weight θ. The GPS point g_{t+1} is located near the intersection, which is close to the candidate point c_{t+1}^1 and c_{t+1}^2, and the distance d_{t+1}^1 is the same as d_{t+1}^2. Connecting g_t and g_{t+1}, the vehicle heading weight between the connecting line and the two segments is θ_1 and θ_2. Considering the impact of vehicle heading weight, c_{t+1}^2 has a greater probability of observation, and we can suppose that c_{t+1}^2 is the real position of the vehicle at time $t+1$.

Transition Probability

In first-order HMM, the transition probability measures the transition from one hidden state to another (*26*). The map matching algorithm based on HMM uses the transition probability to measure the probability of moving from a candidate point c_{t-1}^{i} at time *t*-1 to a candidate point c_t^j at time *t* (*23*). The formula for calculating the transition probability of the first-order HMM in this paper is given as below (*23*):

$$P\left(c_t^i \mid c_{t-1}^i\right)=\begin{cases} p_{same}\dfrac{1}{\beta}e^{-\frac{s_t}{\beta}},\ c_t^i \text{ and } c_{t-1}^i \text{ are on the same/adjacent road segments} \\ \left(1-p_{same}\right)\dfrac{1}{\beta}e^{-\frac{s_t}{\beta}},\ \text{otherwise} \end{cases} \tag{5}$$

where $P\left(c_t^i \mid c_{t-1}^i\right)$ is the transition probability from candidate point c_{t-1}^i to candidate point c_t^j. p_{same} (>0.5) is a parameter. With Equation (5), we can get the transition probability considering if c_{t-1}^i and c_t^j are on the same or adjacent road segments. In this way, the topological relation of road segments is taken into account. β is the mean of s_t. s_t is the difference between the large circle distance from g_{t-1} to g_t (denoted as $dist\left(g_{t-1},g_t\right)$) and the route length from c_{t-1}^i to c_t^j (denoted as $routeDist\left(c_{t-1}^i,c_t^j\right)$):

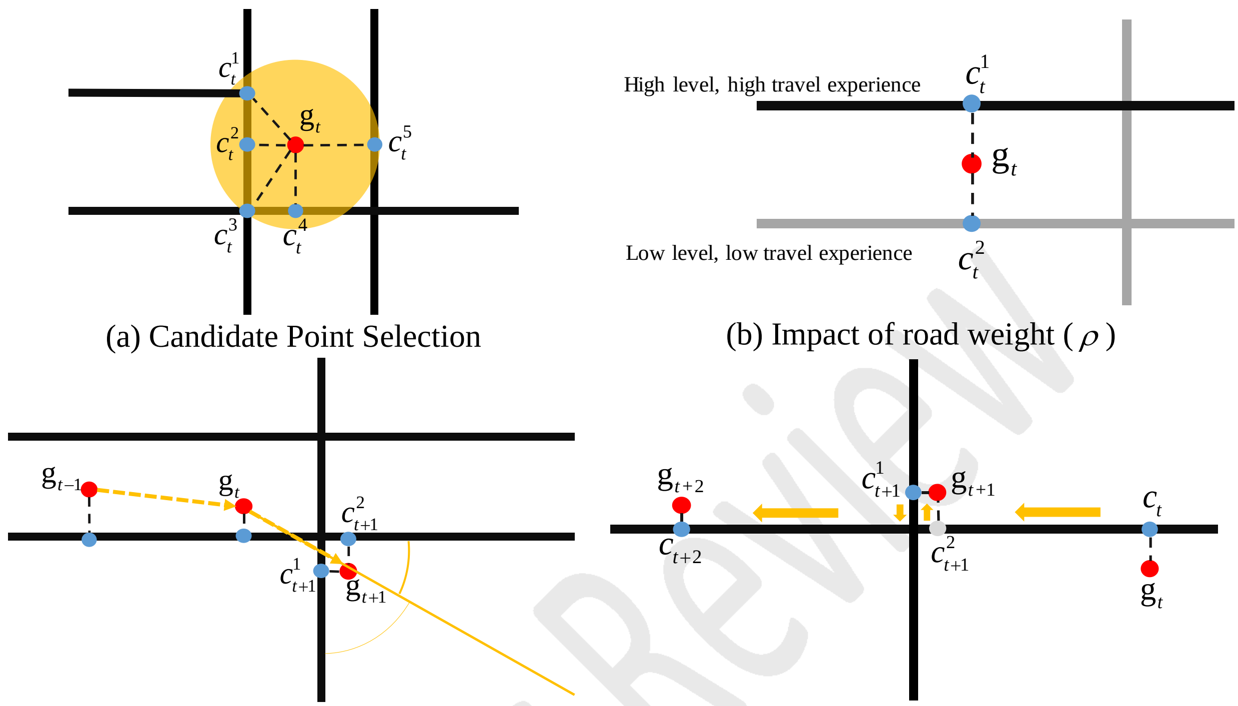
$$s_t=\left|dist\left(g_{t-1},g_t\right)-routeDist\left(c_{t-1}^i,c_t^j\right)\right|. \tag{6}$$

Self-adaptive Sliding Window and Second-Order Transition Probability

Existing first-order HMM online map matching algorithms usually only focus on one single GPS point, considering its local geometric relation and road topology relationship, which results in the precision of online map matching algorithm far behind the higher-order map matching algorithm.

Figure 1(d) shows an example that the conventional first-order HMM online map matching results in incorrect match. Obviously, from GPS point g_t to g_{t+2}, the vehicle does not turn and the correct matching path should be $c_t \to c_{t+1}^2 \to c_{t+2}$. However, in the process of first-order HMM online incremental matching, an incorrect matching result is $c_t \to c_{t+1}^1 \to c_{t+2}$. The reason for this error is that the first-order HMM only considers the observation probability of a single point and the transition probability between two points, so that lacks the measurement of transition probability on a larger scale. However, the real location of the current GPS point is not just related to the previous point, but to multiple previous points.

Higher-order HMM is an extension of HMM (*28*). The basic assumption of higher-order HMM is that the current state is not only related to one previous state, but multiple previous states. In some cases, the higher-order HMM is more consistent with the real situation, such as natural language processing (*29*), speech recognition (*30*) and so on. In the problem of map matching, because the vehicle movement is continuous, the real position of the current point is not only related to the previous point, but also to the trajectory formed by two or more points. Therefore, higher-order HMM is more suitable for map matching than traditional first-order HMM. Analogous to human eyes observing things, we should first pay attention to the characteristics of things as a whole. For example, the connection from g_t to g_{t+2} is approximately a straight line, so the GPS point g_{t+1} is more likely to be matched to c_{t+1}^{2} than c_{t+1}^{1}. To overcome this error and improve the accuracy of online map matching, in this study, we introduce a self-adaptive sliding window and extend the first-order HMM map matching to a second-order one.



(a) Candidate Point Selection

(b) Impact of road weight (ρ)

(c) Impact of vehicle heading weight (θ)

(d) Incorrect match with first-order HMM

Figure 1 Illustration of merits of the proposed map matching method

In the applications such as real-time navigation and travel time estimation, online map matching is necessary. The existing HMM map matching algorithms usually use sliding window to realize online matching. Denote the sliding window size as w (no. of GPS points). If the window overflows after the current point g_t entering the window, the first point in the window g_{t-w} is removed, and the matching result of g_{t-w} point will be finally determined. As the new point continues to join, matching results within the window may be changed continuously. The introduction of sliding window makes online map matching possible, but it is difficult to determine the window size w. If w is too large, the matching speed will be too slow to meet the real-time performance requirement. If w is too small, the matching accuracy will be compromised. To solve this problem, a self-adaptive sliding window is proposed in this study.

We define self-adaptive sliding window size w to three levels, i.e. *small*, *medium* and *large*. By calculating the average value of GPS points positioning error in the current window, sliding windows of different sizes are automatically selected to improve the accuracy of the online map matching as much as possible. The average value of GPS points positioning error (denoted as E_{ave}) can be obtained as

$$E_{ave} = \frac{\sum_{n=t-w+1}^{n=t} dist(g_n, c_n)}{w}, \tag{7}$$

where c_n is the candidate point which is matched to g_n.

Observation probability of second-order HMM $P\left(g_{t-1}, g_t | c_{t-1}^i, c_t^j\right)$ can be obtained from the first-order HMM:

$$P\left(g_{t-1}, g_t | c_{t-1}^i, c_t^j\right) = P\left(c_t^j | c_{t-1}^i\right) \cdot P\left(g_{t-1} | c_{t-1}^i\right) \cdot P\left(g_t | c_t^j\right). \tag{8}$$

Define the second-order HMM state transition probability (denoted as $P\left(c_t^i \mid c_{t-2}^i, c_{t-1}^i\right)$) as

$$P\left(c_t^i | c_{t-2}^j, c_{t-1}^k\right) = \frac{1}{\lambda} e^{-\frac{k_t}{\lambda}}, \tag{9}$$

where λ is the mean of k_t. k_t is the difference between the large circle distance from g_{t-1} to g_{t+1} and the route length from c_{t-1}^i to c_{t+1}^j:

$$k_t = \left| \sum_{n=t-2}^{n=t-1} dist\left(g_n^i, g_{n+1}^j\right) - \sum_{n=t-2}^{n=t-1} routeDist\left(c_n^i, c_{n+1}^j\right) \right|. \tag{10}$$

The second-order transition probability describes the state transition between three consecutive candidate points, that is, the actual position of the current GPS point is related to the previous two points. In this way, the strong assumption of first-order HMM is relaxed and the accuracy of map matching is improved. In fact, we can continue to extend the proposed method to third-order HMM and define appropriate observation and transition probabilities to improve accuracy. However, three-order HMM will make the calculation process more complicated, which is not conducive to online map matching.

Extended Viterbi Algorithm

In the previous sections, we introduce the second-order HMM to solve the map matching problem. Although we use sliding window to reduce the computational complexity of matching a single GPS point, the algorithm complexity of traversing the second-order HMM is still $O(n^w)$. Traversal search seriously affects the online performance of the matching algorithm. Thus, some dynamic programming algorithms should be used to reduce the complexity.

The objective function of second-order HMM dynamic programming is defined as

$$\max \prod_{n=t-w+3}^{n=t} \left(P\left(c_n^i | c_{n-2}^j, c_{n-1}^k\right) \times P\left(g_{n-2}, g_{n-1} | c_{n-2}^j, c_{n-1}^k\right) \right). \tag{11}$$

Viterbi algorithm is an efficient dynamic programming algorithm, which can effectively avoid repeated searches of path and quickly achieve the optimal solution. It is widely used to solve the first-order HMM. To solving the second-order HMM with a complexity of $O(n^2)$, we extend the traditional Viterbi algorithm (*30*) as follows.

Step 1: Order Reduction

In the second-order HMM, $P\left(g_{t-1}, g_t | c_{t-1}^i, c_t^j\right)$ is regarded as the observation probability, which is equivalent to the observation probability of a single candidate point in the first-order HMM. Equation (8) shows that the observation probability of the second-order HMM is the product of the observation probability of two consecutive candidates in the first-order HMM and the state transition probability. Thus, the order of the second-order HMM can be reduced by using Equation (8) (refer to Figure 2(a)). If the second-order HMM has two layers, each layer has m and n nodes respectively, the second-order HMM can be reduced to one layer with $m \times n$ nodes.

Step 2: Recursive Tracing

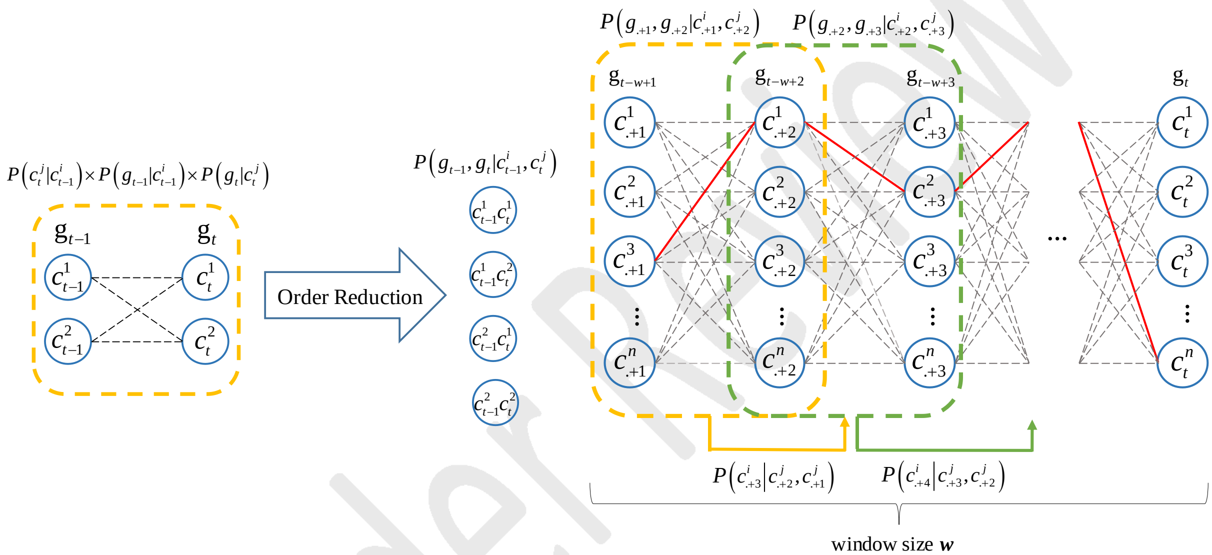
After Step1, we can use the traditional Viterbi algorithm for iterative calculation to solve the second-order HMM in the following process (refer to Figure 2(b)):

a. Starting from the first layer's nodes, the observation probability of each layer's nodes after reduction and the transition probability between adjacent two layers' nodes are calculated.

b. Calculate the maximum total probability of each node from the second layer to the last layer. Save maximum total probability and precursor node of each node.

c. Select the node with the highest total probability in the last layer, and go back to its precursor node until the first layer.

With the above steps, we can find the optimal matching path $\left(c_{t-w+1}^{i}, c_{t-w+2}^{j}, \ldots, c_{t}^{k}\right)$ in the sliding window.



(a) Order reduction process (b) Extended Viterbi algorithm

Figure 2 Illustration of the extended Viterbi algorithm

CASE STUDY

In this section, we make sensitivity analyses of the parameters involved in the algorithm, and use real data to show the merits of the proposed higher-order HMM map matching algorithm.

Data Preparation and Evaluation Metric

We used the road network data of Qinhuai District in Nanjing, China, including 6901 sections and 4647 nodes. Taxi GPS data with 30 seconds sampling interval collected in September 2016 were used. In order to verify the effectiveness of the algorithm under extreme conditions and reflect the advantages of the proposed algorithm, we re-sampled the original data and added the random noise of Gaussian distribution. The re-sampling intervals are 60s to 300s. The Gaussian noises with a standard deviation of 10m to 80m (convert to degrees) were added to longitude and latitude.

Evaluation metric is defined as follows. First, we find the common matching sequence X (the sequence that matched correctly) between the matched output route M and the real trajectory T. Based on this sequence, the precision and the recall of the map-matching result (denoted as *pcs* and *rc* respectively) can be calculated as

$$pcs = \frac{X}{M}, \tag{12}$$

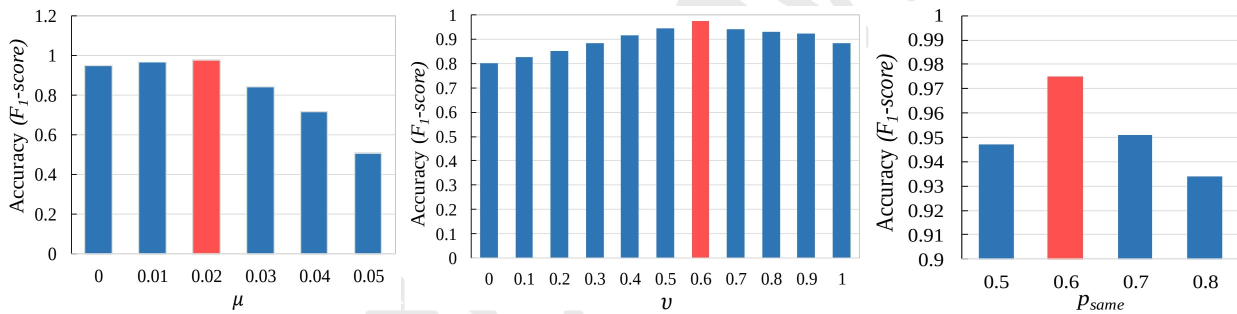
$$rc = \frac{X}{T}. \tag{13}$$

pcs is defined as the ratio of the length of matched sequence *X* and the total length of the matched trajectory *M*. *rc* is defined as the ratio between the length of matched sequence *X* and the total length of the real trajectory *T*. In this study, F_1-*score*, which is widely used to evaluate the performance of classification models and prediction models (*31*), is adopted in this study to evaluate the proposed model:

$$F_1\text{-}score = \frac{2 \cdot pcs \cdot rc}{pcs + rc}. \tag{14}$$

Results

Effects of different parameters on map matching accuracy are investigated in this study. In the proposed model, there are three parameters to be estimated, i.e. μ, υ and p_{same}. According to previous studies, the approximate range of the three parameters can be obtained. Figure 3 shows the impact of different parameter values on F_1-*score* and Table 2 shows the optimal parameter values.



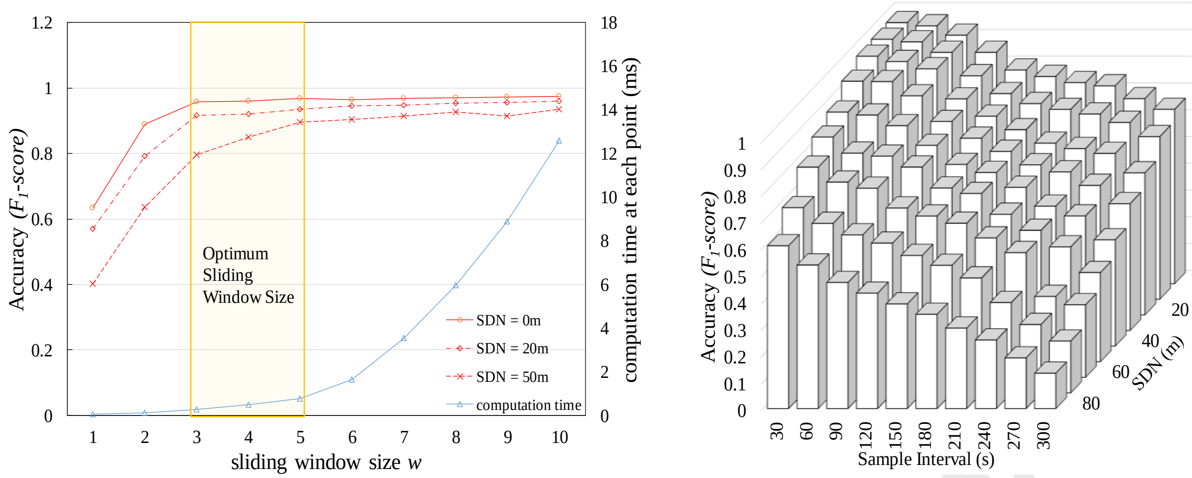
(a) The effect of μ (b) The effect of υ (c) The effect of p_{same}

Figure 3 Sensitivity analyses of model parameters

Table 2 Optimal Parameter Values

Parameter	Optimal Value	Range	Function
μ	0.02	0-0.05	Control the effect of road weight on results
υ	0.6	0-1	Control the effect of vehicle heading weight on results
p_{same}	0.6	0.5-0.8	Control the effect of same/adjacent road priority on results

Figure 4(a) shows the effect of window size *w* on the accuracy of map matching. It can be seen that when *w* = 3, the value of F_1-*score* increases significantly. The reason is that when the size of sliding window is larger than 3, the second-order HMM comes into play. Under different standard deviations of noise (SDNs), when the sliding window size increases from 3 to 10, the matching accuracy gradually becomes unchanged. However, as the sliding window size increases, the computation time of matching a single GPS point increases rapidly. Thus, the optimal three levels of self-adaptive sliding window size are 3, 4 and 5.



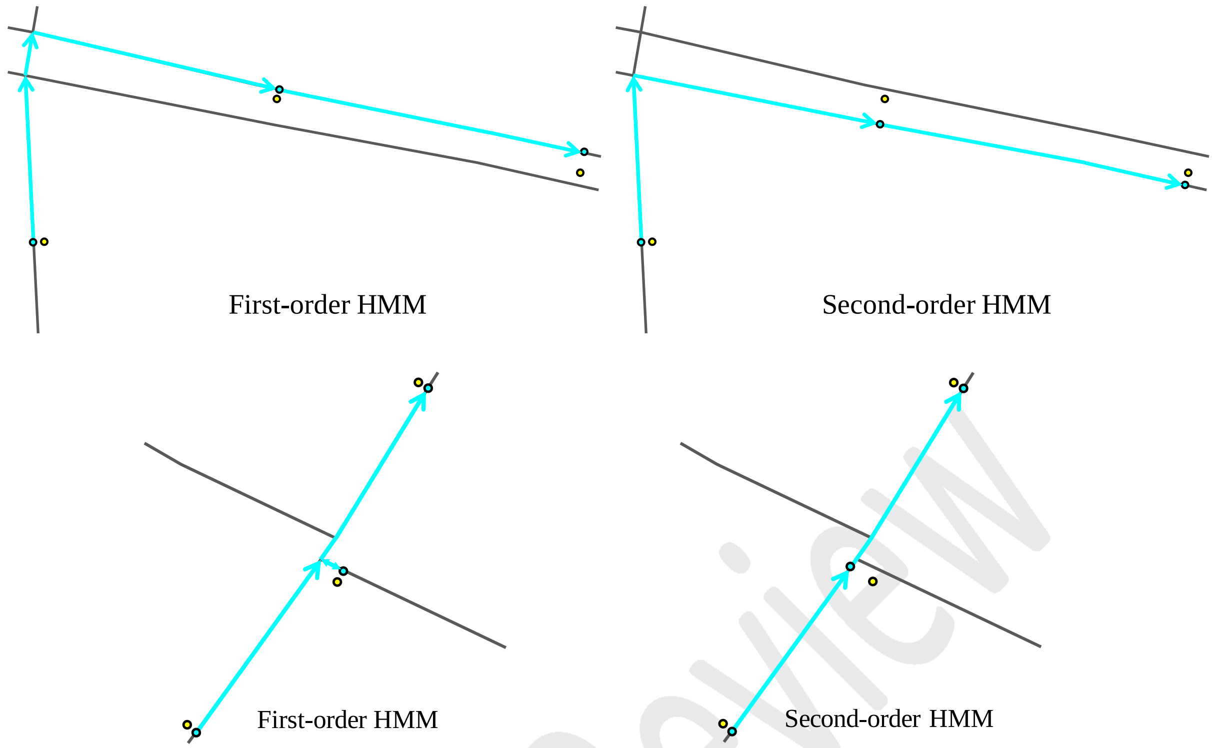
(a) Effect of sliding window size (b) Effects of sample interval and SDN

Figure 4 Effects of sliding window size and sampling on map matching accuracy

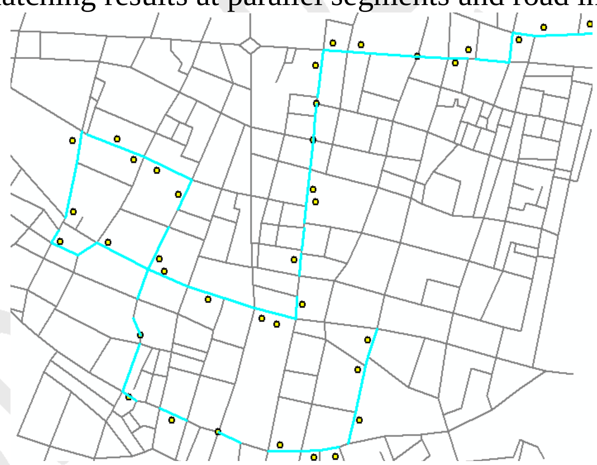
Figure 4(b) shows the effects of the sample interval and the random SDN on accuracy of map matching. With the increases of sampling interval and SDN, the *F1-score* decreases. It can be seen from Figure 4(b) that when the sampling interval is between 30s to 90s and the SDN ranges from 0 to 30 m, the *F1-score* is kept above 0.9.

With the map matching algorithm proposed in this paper, various factors (i.e. road level, driver's travel preference, network topology, vehicle heading and same/adjacent road priority) are considered. Figure 5(a) shows some map matching results in complex urban road network environment. From Figure 5(a), it can be seen that the first-order HMM map matching algorithm may bring mismatch when it deals with parallel road segments. Under the constraints of topological relations, second-order HMM algorithm gives a greater transition probability to the segment which is adjacent to the previous segment to effectively reduce errors. When GPS points are near road intersections, first-order HMM algorithm may match GPS points to the segment crossed the current road. Second-order HMM and sliding window can help solve this problem. The second-order transition probability can effectively avoid the detour of matching trajectory at the intersection and improve the accuracy of map matching. Figure 5(b) shows an overview of map matching result in central area of Nanjing, where the road network is dense and complex. The proposed algorithm is found well performed.

Figure 6(a) compares the performance of the proposed second-order HMM map matching algorithm with the performance of conventional first-order HMM algorithm at different sample intervals without adding random noise. It can be seen that the *F_1-score* of the proposed algorithm is higher than that of the first-order algorithm. With the increase of sampling interval, the advantages of the proposed algorithm become evident. Taking the 300 seconds sampling interval as an example, the distance between two GPS points is about 2500 meters considering the average speed 30km/h on urban roads. In this situation, the position correlation between two consecutive GPS points is very low. The traditional first-order HMM algorithm only considers the transition probability between two points, so the error tends to be very large. Our proposed algorithm integrates several factors such as road level and driver's travel preference, and second-order transition probability can match GPS trajectory on a larger scale, so it shows higher accuracy (*F_1-score* is about 0.67).



(a) Map matching results at parallel segments and road intersections

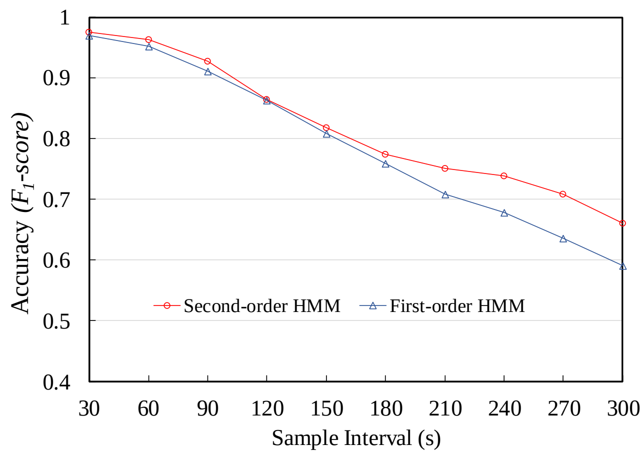


(b) An overview of map matching result

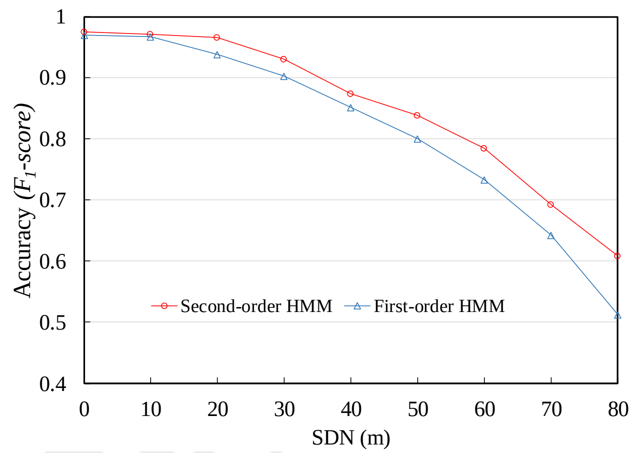
Figure 5 Results of map matching

Figure 6(b) compares the performance of the proposed second-order HMM map matching algorithm with conventional first-order HMM algorithm at different SDNs with 30s sample interval. The map matching accuracy of the proposed algorithm is always higher than that of first-order algorithm. The reason is that the conventional first-order HMM algorithm only considers the distance factor when calculating the observation probability of candidate points. When the positioning error of GPS point increases and the road network is dense, matching errors are numerous. In practical, the GPS positioning error is significant in city centre with intensive high-rise buildings. As the proposed second-order HMM algorithm excelled conventional algorithms in accuracy (0.6 compared to 0.5 when SDN equals 80m), the proposed algorithm can be adopted to achieve high accuracy of map matching in the whole city.

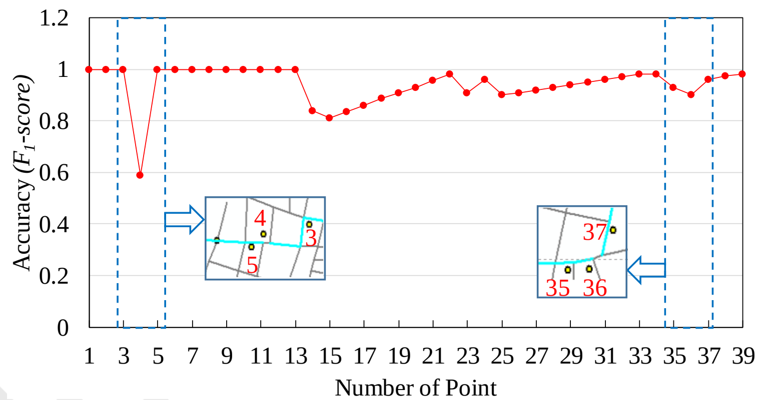
Figure 6(c) shows the dynamic error correction of second-order HMM in online map matching. Taking point 4 as an example, when point 4 is added into the sliding window, a matching error occurs and the accuracy of the algorithm decreases dramatically. However, with the addition of point 5, under the joint action of second-order HMM and sliding window, the matching error of point 4 is dynamically corrected by the proposed algorithm. With the continuous operation of the algorithm, the matching accuracy eventually stabilizes at a high level.



(a) Accuracy at different sampling intervals



(b) Accuracy at different SDNs



(c) Accuracy fluctuation during map matching process

Figure 6 Analysis of map matching accuracy

CONCLUSIONS

An online map matching algorithm based on second-order HMM is presented in this paper. Various factors (i.e. road level, driver's travel preference, road topology, vehicle heading and same/adjacent road priority) are explicitly considered in the algorithm, which effectively improve the accuracy of map matching in complex urban road network environment. An extended Viterbi algorithm is developed to solve the map matching problem efficiently. A self-adaptive sliding window mechanism is proposed to adjust window size on a real time basis and ensures high accuracy. We tested the proposed algorithm using real road network and massive taxi GPS data collected in Nanjing, China. The proposed map matching approach was found outperform state-of-the-art algorithms built on the first-order HMM in various testing environment. The proposed algorithm can be used in real-time navigation, trajectory monitoring, traffic flow analysis and other related fields.

The incorporation of individuals' activity and travel choice behaviour is suggested as an interesting extension of the proposed algorithm, potentially improving the accuracy of map matching (*24*, *32*, *33*). In the cases study, the proposed algorithm used only a single processor. How to best incorporate the cloud and parallel computing technologies into the proposed algorithm needs further investigation (*34*).

ACKNOWLEDGMENTS
The work described in this paper was jointly supported by the National Natural Science Foundation of China (71601045), the Natural Science Foundation of Jiangsu Province (BK20160676) and Six Talent Peaks Project of Jiangsu Province (RJFW-006).

AUTHOR CONTRIBUTIONS
The authors confirm contribution to the paper as follows: study conception and design: Xiao Fu, Biyu Chen, Zhiyuan Liu; data collection: Zhiyuan Liu; analysis and interpretation of results: Jiaxu Zhang, Yue Zhang, Xiao Fu; draft manuscript preparation: Xiao Fu, Jiaxu Zhang, Biyu Chen. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

1. Zou, D., Z. Deng, Z., L. Xu, and W. Ren. Seamless LBS Based on the Integration of WSN and GPS. *International Symposium on Computer Science & Computational Technology*, 2008. DOI: 10.1109/WICOM.2009.5303191.
2. Huang, W., W. Jia, J. Guo, B. M. Williams, G. Shi, Y. Wei, and J. Cao. Real-time Prediction of Seasonal Heteroscedasticity in Vehicular Traffic Flow Series. *IEEE Transactions on Intelligent Transportation Systems*, 2018. 19: 3170–3180.
3. Bao, J., L. Pan, X. Qin, and H. Zhou. Understanding the Effects of Trip Patterns on Spatially Aggregated Crashes with Large-scale Taxi GPS Data. *Accident Analysis & Prevention*, 2018. 120: 281–294.
4. Bierlaire, M., J. Chen, and J. Newman. A Probabilistic Map Matching Method for Smartphone GPS Data. *Transportation Research Part C*, 2013. 26: 78–98.
5. Luo, Q., J. Auld, and V. Sokolov. Addressing Some Issues of Map-matching for Large-scale, High-frequency GPS Data Sets. Presented at 95th Annual Meeting of the Transportation Research Board, Washington, D.C., 2016.
6. White, C. E., D. Bernstein, and A. L. Kornhauser. Some map matching algorithms for personal navigation assistants. *Transportation Research Part C*, 2000. 8: 91–108.
7. Taylor, G., G. Blewitt, D. Steup, S. Corbett, and A. Car. Road Reduction Filtering for GPS‐GIS Navigation. *Transactions in GIS*, 2001. 5: 193–207. DOI:10.1111/1467-9671.00077
8. Velaga, N. R., M. A. Quddus, and A. L.Bristow. Improving the Performance of a Topological Map-Matching Algorithm through Error Detection and Correction. Presented at 91th Annual Meeting of the Transportation Research Board, Washington, D.C., 2012.
9. Chen, B., H. Yuan, Q. Li, W. H. K. Lam, S. L. Shaw, and K. Yan. Map-Matching Algorithm for Large-scale Low-frequency Floating Car Data. *International Journal of Geographical Information Science*, 2014. 28(1): 22–38.
10. Zheng, L., X. Liu, B. Yi. Dynamic Weighted Real-time Map Matching Algorithm Considering Spatio-Temporal Property. *Journal of Computer Applications*, 2017. 37(8): 2381–2386.
11. Liu, X. L., K. Liu, M. X. Li, and F. Lu. A ST-CRF Map-Matching Method for Low-Frequency Floating Car Data. *IEEE Transactions on Intelligent Transportation Systems*, 2017. 18(5): 1241–1254.
12. Gong, Y. J., E. Chen, X. Zhang, L. M. Ni, and J. Zhang. AntMapper: An Ant Colony-based Map Matching Approach for Trajectory-based Applications. *IEEE Transactions on Intelligent Transportation Systems*, 2018. 19(2): 390–401.
13. Takenga, C., T. Peng, and K. Kyamakya. Post-processing of Fingerprint Localization using Kalman Filter and Map-Matching Techniques. Presented at The 9th International Conference on Advanced Communication Technology, 2007.
14. Taguchi, S., S. Koide, and T. Yoshimura. Online Map Matching with Route Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 2018. 1–10.
15. Ren, M. and H. A. Karimi. A Fuzzy Logic Map Matching for Wheelchair Navigation. *GPS Solutions*, 2012. 16(3): 273–282.
16. Chen, Z. and R. C. Qiu. Prediction of Channel State for Cognitive Rradio using Higher-Order Hidden Markov Model. *Proceedings of the IEEE SoutheastCon 2010 (SoutheastCon)*, 2010. DOI: 10.1109/SECON.2010.5453870.

17. Lou, Y., C. Zhang, Y. Zheng, W. Wang, and Y. Huang. Map-Matching for Low-Sampling-Rate GPS Trajectories. *Proceedings of the ACM-GIS*, 2009. 352−361.
18. Yuan, J., Y. Zheng, C. Zhang, X. Xie, and G. Sun. An Interactive-Voting Based Map Matching Algorithm. *Proceedings of Eleventh International Conference on Mobile Data Management*, 2010. DOI: 10.1109/MDM.2010.14.
19. Newson, P. and J. Krumm. Hidden Markov Map Matching through Noise and Sparseness. *Proceedings of ACM SIGSPATIAL*, 2009. 336–343.
20. Bonnetain, L., A. Furno, J. Krug, and N-E. E. Faouz. Can We Map-Match Individual Cellular Network Signaling Trajectories in Urban Environments? Data-Driven Study. *Transportation Research Record: Journal of the Transportation Research Board*, 2019. https://doi.org/10.1177/0361198119847472.
21. Song, R., W. Lu, W. Sun, W. Huang, and C. Chen. Quick Map Matching using Multi-core CPUs. *Proceedings of the ACM-GIS*, 2012. 605−608. DOI:10.1145/2424321.2424428.
22. Atia, M. M., A. R. Hilal, C. Stellings, E. Hartwell, J. Toonstra, W. B. Miners, and O.A. Basir. A Low-cost Lane-Determination System using GNSS/IMU Fusion and HMM-based Multistage Map Matching. *IEEE Transactions on Intelligent Transportation Systems*, 2017. 1–11.
23. Mohamed, R., H. Aly, and M. Youssef. Accurate Real-time Map Matching for Challenging Environments. *IEEE Transactions on Intelligent Transportation Systems*, 2017. 18(4): 847–857.
24. Jagadeesh, G. R., and T. Srikanthan. Online Map-Matching of Noisy and Sparse Location Data with Hidden Markov and Route Choice Models. *IEEE Transactions on Intelligent Transportation Systems*, 2017. 1–12.
25. Salnikov, V., M. T. Schaub, and R. Lambiotte. Using Higher-Order Markov Models to Reveal Flow-based Communities in Networks. *Scientific Reports*, 2016: 6.
26. Eddy, S. R. What is a Hidden Markov Model?. *Nature Biotechnology*, 2004. 22: 1315–1316.
27. Leibowitz, N., B. Baum, G. Enden, and A. Karniel. The Exponential Learning Equation as a Function of Successful Trials Results in Sigmoid Performance, *Journal of Mathematical Psychology*, 2010. 54(3): 338–340,
28. Ye, F. and Y. Wang. Research Advancement of High-Order Hidden Markov Model. *Advances in Mathematics*, 2014. 2(2): 845–848.
29. Lee, T., F. Zheng, W. Wu, and D. Chen. The Hidden Markov Model of Co-articulation and its Application to the Continuous Speech Recognition. *Journal of Electronics (China)*, 2000. *17*(3): 242–247.
30. Gales, M. J. F., S. Watanabe, and E. Fosler-Lussier. Structured discriminative Models for Sspeech Recognition: An Overview. *IEEE Signal Processing Magazine*, 2012. *29*(6): 70–81.
31. Voorhees, E. M. Variation in Relevance Judgments and the Measurement of Retrieval Effectiveness. *Information Processing & Management*, 2000. 36(5): 697–716.
32. Fu, X. and W. H. K. Lam. A Network Equilibrium Approach for Modelling Activity-Travel Pattern Scheduling Problems in Multi-modal Transit Networks with Uncertainty. *Transportation*, 2014. 41: 37–55.
33. Fu, X. and W. H. K. Lam. Modelling Joint Activity-Travel Pattern Scheduling Problem in Multi-modal Transit Networks. *Transportation*, 2018. 45: 23–49.
34. Li, Q., T. Zhang, and Y. Yu. Using Cloud Computing to Process Intensive Floating Car Data for Urban Traffic Surveillance. *International Journal of Geographical Information Science*, 2011. 25: 1303–1322.