Probabilistic map matching of sparse and noisy smartphone location data

Jagadeesh, George Rosario; Srikanthan, Thambipillai

2015

Jagadeesh, G. R., & Srikanthan, T. (2015). Probabilistic map matching of sparse and noisy smartphone location data. Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems, 812-817. doi:10.1109/ITSC.2015.137

<https://hdl.handle.net/10356/145769>

<https://doi.org/10.1109/ITSC.2015.137>

© 2015 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at:
<https://doi.org/10.1109/ITSC.2015.137>

Downloaded on 26 Apr 2023 20:22:09 SGT

Probabilistic Map Matching of Sparse and Noisy Smartphone Location Data

G. R. Jagadeesh

School of Computer Engineering
Nanyang Technological University
Singapore 639798
E-mail: asgeorge@ntu.edu.sg

T. Srikanthan

School of Computer Engineering
Nanyang Technological University
Singapore 639798
E-mail: astsrikan@ntu.edu.sg

Abstract—There is an immense amount of location data being collected today from smartphone users by various service providers. Due to bandwidth and battery-life considerations, smartphone locations are generally sampled at sparse intervals using energy-efficient, but inaccurate, alternatives to the power-hungry Global Positioning System (GPS). If sparse sequences of coarse location data obtained from mobile users can be accurately map-matched to travel paths on the road network, then this data can be effectively used for several traffic-related applications. Unlike most other map-matching methods in the literature, we, in this paper, focus on the problem of map-matching sparse and noisy non-GPS smartphone location data. We adopt the widely-followed Hidden Markov Model (HMM) approach and propose new probabilistic models for the observation and transition probabilities tailored towards the type of data being considered. Our map-matching method has been evaluated using ground-truth labelled non-GPS location data collected from real drives. Tests show that the accuracy of the proposed method is about 12% more than that of a comparable HMM-based method from the literature. Our results also show that the runtime and latency of the proposed method can be kept within reasonable bounds using simple techniques.

Keywords—*map matching, path inference, probabilistic models, location data*

I. INTRODUCTION

A large proportion of smartphone users use location-based services and agree to share their location information with service providers. For instance, Android smartphone users who use Google's Wi-Fi location service allow the collection of anonymous location data even when no applications are running. Location data collected from mobile smartphone users can serve as valuable inputs for applications such as traffic congestion detection and travel time estimation. However, this is predicated on the ability to infer the paths travelled by mobile users from their time-stamped location measurements. This task is known as map matching. It is not a straightforward task because the location data is sparse (due to the typically low sampling rate) and noisy (due to the location measurement error).

Smartphones use multiple sources with varying accuracies such as Global Positioning System (GPS), Wi-Fi positioning and cellular network positioning for location estimation. According to one study, the median errors for location estimates from GPS, Wi-Fi and cellular positioning are 8 m, 74 m and 600 m respectively [1]. While GPS provides superior accuracy, it depletes the phone's battery several times faster than the other

positioning methods [2][3]. The power hungriness of GPS is well known and many smartphone users keep GPS disabled in order to save battery power. Besides, there are smartphones without GPS functionality, especially in price-conscious markets. In order to keep their consumption of the users' bandwidth and battery power at negligible levels, many applications sample the smartphone locations at sparse intervals using energy-efficient, but inaccurate, alternatives to GPS. Map matching such noisy data, especially when sparsely sampled, is difficult compared to map matching GPS data. The former has received very little attention in the literature.

Starting from the early nineties, a large quantum of research has been published on map matching GPS locations [4]. Until about a decade ago, the proposed methods were mostly meant for use in personal navigation devices and dealt with densely-sampled GPS data (typically, one sample per second). The problem of map matching sparsely-sampled GPS data started receiving attention after the emergence of Floating Car Data (FCD) systems that use connected vehicles as traffic probes. In this context, map matching can be seen as the problem of identifying the true travel path of the vehicle from among a set of candidate paths for a given trajectory of location samples. Various approaches to solving this problem have been proposed. These include geometric methods such as using the Fréchet distance to assess the similarity of the vehicle trajectory and the candidate paths [5]. An alternative approach is to use a model of drivers' route choice to select the best path from the set of candidate paths [6][7].

Another class of map matching algorithms use probabilistic techniques to deal with the uncertainties involved in inferring the vehicle's travel path. In particular, Hidden Markov Models (HMM) are well-suited for modelling the candidate road segments corresponding to each location sample and the transitions between candidate road segments of one sample to those of the next. There are a number of HMM-based map-matching methods proposed in the literature [8] [9] [2] [10] [11]. Another probabilistic model, Conditional Random Fields (CRF), has also been used for map matching [12][13]. Almost all the map-matching methods proposed in the literature deal only with GPS data. A notable exception is the work by Thiagarajan et al. [2][9], who explored map matching of location data from Wi-Fi and cellular positioning. However, they used continuous densely-sampled non-GPS location data. In practice, due to bandwidth considerations, location data collected from smartphones are unlikely to be continuous.

In this paper, we focus on the problem of map matching sparsely-sampled and noisy non-GPS location samples. For this, we use HMM, which is the most widely-followed approach for probabilistic map matching. We propose new probabilistic models for the emission and transition probabilities in the HMM. Evaluations conducted with non-GPS location data from real drives show substantial improvement in the map-matching accuracy compared to an influential HMM-based method from the literature. We also evaluate techniques for reducing the runtime and latency of the proposed method.

The rest of the paper is organized as follows. Section II explains how HMM can be applied for map matching. In Section III, we present new models for estimating the emission and transition probabilities in HMM. The preparation of ground truth data and the estimation of probabilistic model parameters from it are described in Section IV. We present the experimental setup and the results in Section V. Section VI concludes the paper.

II. HMM-BASED MAP MATCHING

Let l_t denote a location sample obtained from a vehicle-based mobile user at time step t . Each location sample consists of a latitude, longitude and a timestamp. Let R represent a path in the road network such that $R = (r_k | k = 1, \dots, K)$ is an ordered sequence of K road segments $r_1, \dots, r_K$. Given a sequence of location samples $l_1, \dots, l_T$, corresponding to T time steps, the map-matching task is to find the path R^* , which is most consistent with the given sequence.

In general, map-matching methods associate each location sample with a set of candidate road segments that lie within a predetermined error radius. In the HMM-based approach, each location sample is considered as an observation and each of its candidate road segments is represented as a hidden state. To be more specific, a hidden state represents a candidate point, which is the point on a candidate road segment that is closest to the observed location.

Each hidden state is assigned an emission probability, which represents the conditional probability of the location observation being generated if the candidate point represented by the hidden state is the true location. The probability of a vehicle moving from a candidate point at time step t to another candidate point at time step $t+1$ is given by a transition probability distribution. HMM-based map-matching methods in the literature differ mostly in the way they model the emission and transition probability distributions. In Section III, we present new emission and transition probability models that are tailored for sparse and noisy non-GPS location data.

Fig. 1. A simple HMM lattice showing hidden states and transitions.

The hidden states (candidate points) and the transitions between them form a lattice such as the simplified example shown in Figure 1, where the nodes denote the hidden states and the edges denote the transitions. The transition edge from one candidate point to another represents the best path between them according to a route-planning criteria such as the shortest distance or the minimum travel time. Figure 1 shows three time steps and three candidate points per time step. The unknown true sequence from time step 1 to time step 3 includes one candidate point at each time step. As it can be seen, there are several possible sequences in the lattice. The most probable sequence of candidate points is found using the Viterbi algorithm [14]. The Viterbi algorithm is a dynamic programming algorithm that finds the most likely sequence in the HMM lattice by maximizing the product of the emission and transition probabilities along the sequence.

Some details about our implementation of the above-described HMM approach need to be noted. The number of candidate points considered for each location observation determines the number of shortest path computations needed and has a significant impact on the runtime. To keep the runtime within reasonable bounds, only those road segments that lie within an error radius are considered as candidates. The error radius is determined based on the typical range of the measurement errors for the positioning systems used. However, there is a small chance that for some location observations, the true location may lie much beyond the error radius. This may result in a break in the HMM lattice with no valid transitions from one time step to the next [8]. When such a condition is detected during runtime, we skip the outlier observation and continue with the subsequent observation.

To further reduce the runtime at the possible cost of some reduction in the accuracy, we use an intuitive heuristic proposed in [15]. This heuristic method places a fixed limit, say k , on the number of candidate points per time step. At each time step, only the top k candidate points based on their joint probabilities are retained for further computation. (In this context, the joint probability of a hidden state in the HMM lattice is the product of the emission and transition probabilities along the most likely sequence up to that state.) We refer to this method as the *top-k heuristic* method.

HMM-based map-matching algorithms can be run in two modes, namely, *global* and *online*. Global algorithms find the most likely sequence of candidate points only after all the location observations have been received and processed. Hence, they typically incur a large latency and are unsuitable for applications where timely map matching is needed. Online algorithms output partial sequences that are possibly suboptimal without waiting for all the observations to be received. A simple yet effective way to implement this functionality is to use a sliding window of fixed size, say b , to ensure a fixed latency of b time steps. At any time step t , the algorithm computes the most-likely sequence from time step $t-b$ to time step t and outputs the candidate point at time step $t-b$ that is part of the most-likely sequence. Subsequently sliding window is advanced by one step. We refer to this online method as the *fixed-sized sliding window* method.

III. MODELS FOR EMISSION AND TRANSITION PROBABILITIES

A. Emission Probability

In HMM-based map matching, each candidate point of a given location sample is assigned an emission probability. It represents the probability of observing the location sample if the candidate point is the true location. A location observation closer to a candidate point would generally result in a higher emission probability compared to one that is far away. If the candidate point is the true location, then the distance between it and the observed location is nothing but the measurement error. Map-matching algorithms generally assume that the measurement error follows a single zero-mean Gaussian distribution [8] [9] [10] [11] [12]. However, this assumption is incorrect for non-GPS smartphone location data as explained below.

When the GPS functionality is disabled in a smartphone, it typically estimates the location using a combination of Wi-Fi positioning and cellular network positioning. In built-up urban areas, the phone typically receives signals from a number of Wi-Fi access points, whose locations have already been learned and mapped by Wi-Fi location service providers such as Google. In such situations, the phone uses Wi-Fi positioning and the location measurement error is generally within about 150 m [3]. However, when the phone is in an open area with no buildings nearby, it generally relies solely on cellular network positioning, whose location measurement errors can be as large as a few kilometers. Thus, each non-GPS location sample from a smartphone is generated using one of the two positioning systems with substantially different error distributions.

We assume that the measurement errors from Wi-Fi and cellular network positioning systems follow two different zero-mean Gaussian distributions. The location samples originate from either of these two distributions. Hence, we propose that it is best to model the location measurement errors using a two-component zero-mean Gaussian mixture distribution.

Let us denote the j^{th} candidate point at time step t corresponding to an observed location l_t as $s_{t,j}$. Let $d_{t,j}$ be the distance between l_t and $s_{t,j}$. If $s_{t,j}$ is the true location, then $d_{t,j}$ is the measurement error of the observed location. Assuming that the measurement errors follow a two-component zero-mean Gaussian mixture distribution, the emission probability of candidate point $s_{t,j}$ for observation l_t is given as:

$$G(s_{t,j}, l_t) = \frac{w_1}{\sigma_1 \sqrt{2\pi}} e^{-\frac{d_{t,j}^2}{2\sigma_1^2}} + \frac{w_2}{\sigma_2 \sqrt{2\pi}} e^{-\frac{d_{t,j}^2}{2\sigma_2^2}} \quad (1)$$

where w_i and σ_i are weight and standard deviation, respectively, of the i^{th} component distribution. Given a set of n location observations $\{l_1, \dots, l_n\}$ whose corresponding true states $\{s_1^*, \dots, s_n^*\}$ and measurement errors $\{d_1^*, \dots, d_n^*\}$ are known, the parameters w_1, w_2, σ_1 and σ_2 that maximize the likelihood function $\prod_{i=1}^n G(s_i^*, l_i)$ can be estimated using the well-known Expectation Maximization (EM) algorithm as described in [16].

B. Transition Probability

We compute the transition probability between a candidate point $s_{t,j}$ at time step t to another candidate point $s_{t+1,k}$ at time step $t+1$ based on the features of the best path between them. Here, we use the term "best path" to denote the path with the minimum estimated travel time. (Our analysis of real drive data shows that

paths with minimum travel time are overwhelmingly favored.) For estimating the transition probability, an effective path feature is the absolute difference between the length (driving distance) of the path and the straight-line distance between the location observations corresponding to the candidate points [8][13]. Newson and Krumm [8] modelled the length difference as an exponential probability distribution, which gives the transition probability. The underlying idea is that direct paths with smaller length differences are more likely to be true paths compared to circuitous paths with larger length differences. We use this method with the following changes.

In the case of noisy non-GPS data, the distance between the observed locations may not be a reasonably accurate measure. Instead, we use the distance between the candidate points themselves. Also, we take the sampling interval into consideration while evaluating the length difference. For a path q_i between candidate point $s_{t,j}$ and candidate point $s_{t+1,k}$, we define a new length difference measure as

$$y_i = \frac{D_d(s_{t,j}, s_{t+1,k}) - D_s(s_{t,j}, s_{t+1,k})}{\Delta t} \quad (2)$$

where Δt is the sampling interval in minutes and the functions D_d and D_s give the driving distance and the straight-line distance, respectively, between the candidate points. Analysis of ground truth data (explained in the next section), shows that the y_i values roughly follow an exponential distribution given by

$$p(y_i) = \lambda_1 e^{-\lambda_1 y_i} \quad (3)$$

where the parameter λ_1 is estimated from ground truth data.

Besides the length difference measure described above, we also consider a measure of travel time difference. A direct way of doing this would be to use the difference between the path's estimated travel time and the actual travel time (i.e., the elapsed time between samples) [17]. However, this is not a robust approach as the actual travel times can differ considerably from the estimated travel times due to traffic conditions.

We calculate the travel time differences in such a way that differences caused by traffic conditions are ignored. As a simple example, let us consider a path with an estimated travel time of 2 minutes based on the speed limits along it. It is possible for the actual travel time on this path to be 5 minutes due to traffic congestion. However, it would be almost impossible for the actual travel time on this path to be 20 seconds, as it would require driving at a speed several times higher than the speed limits. Based on the above insight, we define the travel time difference measure for a path q_i as

$$z_i = \frac{\max((T_e(q_i) - \Delta t), 0)}{\Delta t} \quad (4)$$

where $T_e(q_i)$ is the estimated travel time of the path q_i and Δt is the elapsed time between samples. The travel time differences are assumed to follow an exponential distribution given by

$$p(z_i) = \lambda_2 e^{-\lambda_2 z_i} \quad (5)$$

where the parameter λ_2 needs to be estimated.

Having defined the above two probability distributions, we specify the transition probability associated with a path q_i as

$$H(q_i) = p(y_i)p(z_i) \quad (6)$$

IV. PREPARATION AND ANALYSIS OF GROUND TRUTH DATA

The non-GPS smartphone location data used in this work were recorded during 10 taxi trips taken in Singapore. The duration of the trips ranged from 7 minutes to 33 minutes and totaled 159 minutes. The total distance travelled was 104 km out of which 52 km (50%) was on expressways. In all the trips, the taxi drivers were given a destination and let to decide the route themselves. An application on an Android smartphone with the GPS disabled was used to record the locations at the fastest possible rate (once every one or two seconds). A total of 6414 location points were collected.

In order to determine the ground truth locations corresponding to the location samples collected, we used another Android smartphone with the GPS enabled to record relatively-accurate, high-frequency GPS data (one sample per second). The phone was placed near the windshields of the vehicles to obtain a good view of the GPS satellites for maximum accuracy. The dots in the top half of Figure 2 are the noisy non-GPS location points collected during a part of one of the trips. The bottom half shows the corresponding GPS trace. We used a variant of the method described in [11] to map-match the GPS locations and infer the actual travelled paths. These paths were subsequently manually verified through visual inspection. Thus, for every non-GPS location point collected, we know the ground truth location and the measurement error.

The histogram of the location measurement error is shown in Figure 3. For visual clarity, the horizontal axis is limited to 1000 m although a small proportion of data points have more than 1000 m error. 99.9% of the errors are within 1500 m. Hence, we set 1500 m as the error radius used for forming the set of candidate points for each location observation. A very small number (9 out of 6414) of errors exceed 1500 m, with the worst case being 8116 m. While some of these gross errors could possibly be due to the relocation of reference Wi-Fi access points [3], most of them appear to be caused by the phone receiving signal from a far-away cell tower in vast open areas.

If the location measurement errors are assumed to follow a single Gaussian distribution, then the standard deviation estimated by maximum likelihood, σ_m , is 257.17 m for our dataset. Many map-matching methods rely on an alternative, robust method based on the Median Absolute Deviation (MAD) for estimating the standard deviation without being unduly affected by outliers [8][10]. For a given set of location measurement errors given by $D = (d_i | i = 1, \dots, N)$, the standard deviation estimated through the robust MAD method is

$$\sigma_r = 1.4826 \text{ median}_i(d_i) \quad (7)$$

For our dataset, σ_r is 82.82 m. The substantial difference in the values of the conventional and robust estimates of the standard deviation suggests that the location measurement errors do not conform to a single Gaussian distribution.

The parameters of the Gaussian mixture distribution in (1) were estimated using the EM algorithm. The weights of the two component distributions were initialized by setting both weights as 0.5 (so that $w_1 + w_2 = 1$). The above given values of σ_m and σ_r , respectively, were used to initialize the standard deviations of the two component distributions. After 10 iterations, the EM algorithm converged to the parameter values shown in Table I.

Fig. 2. Non-GPS locations (top) with the corresponding GPS trace (bottom).

As described in Section III-B, the transition probability depends on the length difference measure defined in (2) and the travel time difference measure defined in (4). We sampled the ground truth data with the sampling intervals ranging from 1 minute to 5 minutes and studied the length differences of the true paths between consecutive sample points. The histogram of the length difference measure is shown in Figure 4. Assuming an exponential distribution, the parameter λ_1 in (3) can be estimated by the robust method used in [8]. Given a set of true length differences $Z = (z_i | i = 1, \dots, N)$, the parameter λ_1 can be estimated as

$$\lambda_1 = \frac{\ln(2)}{\text{median}_i(z_i)} \quad (8)$$

Using the ground truth data, we obtained a value of $\lambda_1 = 0.0113$.

As for the travel time difference measure, the paths sampled from the ground truth data did not have many instances where the actual travel time was less than the travel time estimate based on the speed limits. (Although overspeeding is a commonly observed behavior, we believe that taxi drivers with passengers are less likely to engage in it.) Due to the lack of adequate data, we did not estimate the parameter λ_2 in (5) from the ground truth data. Instead, we empirically set the value of λ_2 as 5.

Fig. 3. Histogram of the location measurement error.

Fig. 4. Histogram of the length difference measure.

TABLE I. ESTIMATED PARAMETERS OF THE GAUSSIAN MIXTURE DISTRIBUTION

Parameter	Component distribution 1	Component distribution 2
Weight	0.21	0.79
Standard deviation (m)	545.16	73.03

V. EVALUATION

A. Experimental Setup and Evaluation Criteria

We subsampled the non-GPS location data at sampling intervals ranging from 30 seconds to 5 minutes. The first sample point was randomly chosen within the first minute of the trip. To ensure fair comparison between different methods, the same set of sample points were used for all the runs of experiments. Given a sequence of sample location points, the map matching method was tasked to infer the travelled path, as a sequence of road segments, from the first point to the last.

The map-matching accuracy was determined by comparing the inferred path with the known ground truth path for all the trips. We used the F-score, which is the harmonic mean of precision and recall, as the measure of accuracy. In the context of map matching, precision and recall are defined as

$$\text{precision} = \frac{L(\{\text{correctly identified segments}\})}{L(\{\text{identified segments}\})} \quad (9)$$

$$\text{recall} = \frac{L(\{\text{correctly identified segments}\})}{L(\{\text{true segments}\})} \quad (10)$$

where $\{\text{true segments}\}$ is the set of road segments in the ground truth path, $\{\text{identified segments}\}$ is the set of road segments in the path inferred by the map-matching method, $\{\text{correctly identified segments}\}$ is the set of road segments in the inferred path that agree with the ground truth and the function L gives the total length of all the road segments in a given set.

B. Alternative Method Used for Comparison

We compared the accuracy of our method with the well-cited method proposed by Newson and Krumm [8]. Their solution, like ours, is based on the HMM approach and targets sparse and noisy location data. It models the location measurement error as a single zero-mean Gaussian distribution. It assumes that the shortest-distance path is taken between candidate points of successive time steps. For the transition probability, Newson and Krumm used the conventional length difference feature described in the beginning of Section III-B.

C. Results

A comparison of accuracy (the F-score expressed as a percentage) between the proposed and compared map-matching methods is shown in Figure 5. For each of the sampling intervals considered, our solution outperforms the Newson-Krumm method by a significant margin. Aggregating the results for all the sampling intervals, the proposed method achieves an overall map-matching accuracy of 88.79%, while the corresponding value for the Newson-Krumm method is 76.71%.

When all the sampling intervals are considered, our test data has an average of 10 location samples per trip. On average, there are 1147 candidate points per location sample leading to a large amount of shortest-path computations between candidate points of successive stages. This causes the runtime of the map-matching method to be unacceptably high. The average amount of time taken to map-match one trip was found to be 112 seconds on a computer with a 3.2 GHz CPU and 16 GB physical memory. In order to limit the runtime to a reasonable value, we evaluated the top- k heuristic method described in Section II. The overall accuracies for various values of k (i.e., the limit on the number of candidate points) are shown in Figure 6. The last column in the graph corresponds to the case where no limit is placed on the number of candidate points (i.e., $k = \infty$). The accuracy of the top- k heuristic method matches that of the non-heuristic method for $k \geq 55$. It can be seen from Figure 7, that the top- k heuristic method succeeds in reducing the runtime to a small fraction of the runtime of the non-heuristic method, while maintaining the same level of accuracy.

We evaluated the suitability of the proposed method for low-latency online operation by applying the fixed-sized sliding window method described in Section II. The accuracies for various window sizes are shown in Figure 8. A window size of b means that the online method has a fixed latency of b time steps. The accuracy of the online method increases with the window size until the window size is 3. The accuracies do not converge to an exact value for higher window sizes because for a given window size b , the inferred paths corresponding to the last $b-1$ samples of each trip will not be generated by the online method. This causes the total length of the paths considered for the evaluation to be slightly different for different window sizes. It can be concluded from the results that the sliding-window based online method with a window size of 3 achieves almost the same accuracy as the global method.

Fig. 5. Accuracy of the proposed and compared methods.

Fig. 6. Accuracy of the top- k heuristic method for various k values.

Fig. 7. Runtime of the top- k heuristic method for various k values.

Fig. 8. Accuracy of the online map-matching method

VI. CONCLUSIONS

This work was motivated by the belief that if the massive amount of discontinuous, coarse location data that are being shared by smartphone users can be map-matched in an accurate and timely manner, it would lead to improved traffic and travel time estimation. We used the HMM-based map-matching framework and proposed improved probabilistic models for the emission and transition probabilities in the HMM. The proposed map-matching solution has been tested on non-GPS location data collected from real drives along with associated ground truth. Results indicate that for the type of location data targeted, our method is capable of achieving substantially higher accuracy compared to the state of the art. We also successfully applied simple techniques from the literature to considerably reduce the runtime and latency of the proposed method.

As part of future work, we intend to validate the ideas presented in this paper on a larger dataset containing location

data from a variety of road users in order to eliminate any biases that may be present in our test data collected solely from taxi trips. For instance, taxi drivers are known to be adept in choosing optimal routes and are less likely to take circuitous paths compared to the general motorist population. It is worth evaluating if the proposed method works equally well in the presence of diverse route choice behavior.

REFERENCES

- [1] P. A. Zandbergen, "Accuracy of iPhone locations: A comparison of assisted GPS, WiFi and cellular positioning", *Transactions in GIS*, vol. 13(s1), pp. 5-25, 2009.
- [2] A. Thiagarajan, L. Ravindranath, H. Balakrishnan, S. Madden, and L. Girod, "Accurate, low-energy trajectory mapping for mobile devices", In Proceedings of the 8th USENIX Conference on Networked Systems Design and Implementation (NSDI), 2011.
- [3] Y. Tsuda, Q. Kong, & T. Maekawa, "Detecting and correcting WiFi positioning errors", In Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 777-786, 2013.
- [4] M.A. Quddus, W.Y. Ochieng, and R.B. Noland, "Current map-matching algorithms for transport applications: State-of-the art and future research directions", *Transportation Research Part C: Emerging Technologies*, vol. 15(5), pp. 312-328, 2007.
- [5] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk, "On map-matching vehicle tracking data", In 31st International Conference on Very Large Data Bases, pp. 853-864, 2005.
- [6] M. Rahmani, and H. N. Koutsopoulos, "Path inference from sparse floating car data for urban networks", *Transportation Research Part C: Emerging Technologies*, vol. 30, pp. 41-54, 2013.
- [7] T. Miwa, D. Kiuchi, T. Yamamoto, and T. Morikawa, "Development of map matching algorithm for low frequency probe data", *Transportation Research Part C: Emerging Technologies*, vol. 22, pp. 132-145, 2012.
- [8] P. Newson & J. Krumm, "Hidden Markov map matching through noise and sparseness", In Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 336-343, 2009.
- [9] A. Thiagarajan, L. S. Ravindranath, K. LaCurts, S. Toledo, J. Eriksson, S. Madden, and H. Balakrishnan, "Vtrack: Accurate, energy-aware traffic delay estimation using mobile phones", In 7th ACM Conference on Embedded Networked Sensor Systems (SenSys), 2009.
- [10] C.Y. Goh, J. Dauwels, N. Mitrovic, M. T. Asif, A. Oran, and P. Jaillet, "Online map-matching based on Hidden Markov model for real-time traffic sensing applications", In Proceedings of the 15th IEEE Intelligent Transportation Systems Conference, pp. 776-781, 2012.
- [11] G. R. Jagadeesh and T. Srikanthan, "Robust real-time route inference from sparse vehicle position data", In Proceedings of the 17th IEEE Intelligent Transportation Systems Conference, pp. 296-301, 2014.
- [12] T. Hunter, P. Abbeel, and A. M. Bayen, "The path inference filter: model-based low-latency map matching of probe vehicle data", *IEEE Transactions on Intelligent Transportation Systems*, vol. 15(2), pp. 507-529, 2014.
- [13] J. Yang, and L. Meng, "Feature selection in conditional random fields for map matching of GPS trajectories", In *Progress in Location-Based Services 2014*, Springer International Publishing, pp. 121-135, 2015.
- [14] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", *IEEE Transactions on Information Theory*, vol. 13(2), pp. 260-269, 1967.
- [15] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang, "Map-matching for low-sampling-rate GPS trajectories", in *17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 352-361, 2009.
- [16] L. Trailovic and L. Y. Pao, "Position error modeling using Gaussian mixture distributions with application to comparison of tracking algorithms", In Proceedings of the American Control Conference, vol. 2, pp. 1272-1277, 2003.
- [17] J. Krumm, J. Letchner, and E. Horvitz, "Map Matching with Travel Time Constraints", In Society of Automotive Engineers World Congress, 2007.